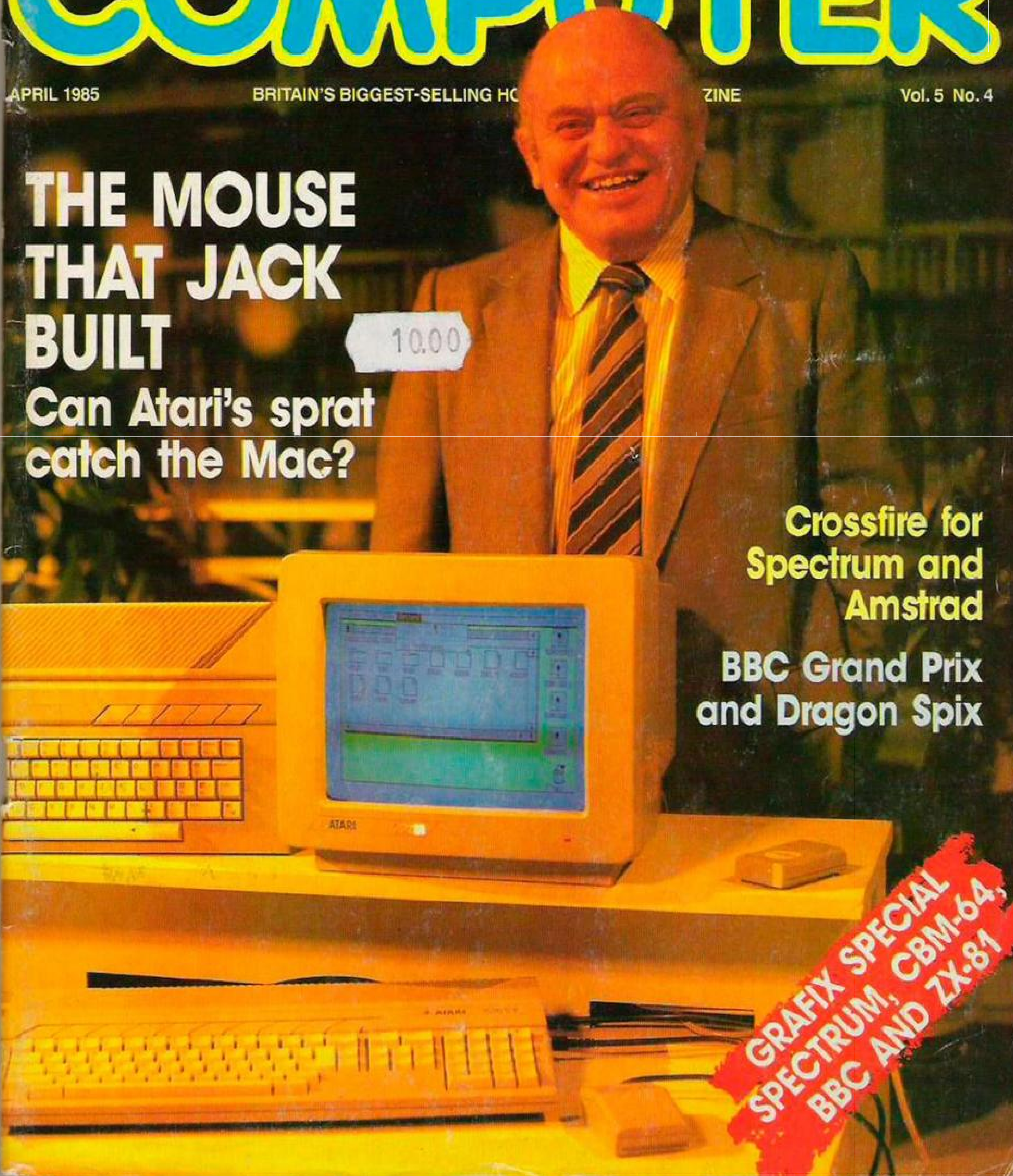
WIN AN ATARI

YOUR COMPUTER

Denmark	DKr.	36.00
France	Fr.	52.50
Germany	DM.	9.50
Greece	Dra.	310.00
Holland	DFl	8.90
Italy	L	4300
New Zealand	NZ$	3.70
Singapore	M$	7.20
Spain	Pts.	400.00
Switzerland	SFr.	7.50

£1

APRIL 1985 BRITAIN'S BIGGEST-SELLING HO[ME COMPUTER MAGA]ZINE Vol. 5 No. 4



THE MOUSE THAT JACK BUILT

Can Atari's sprat catch the Mac?

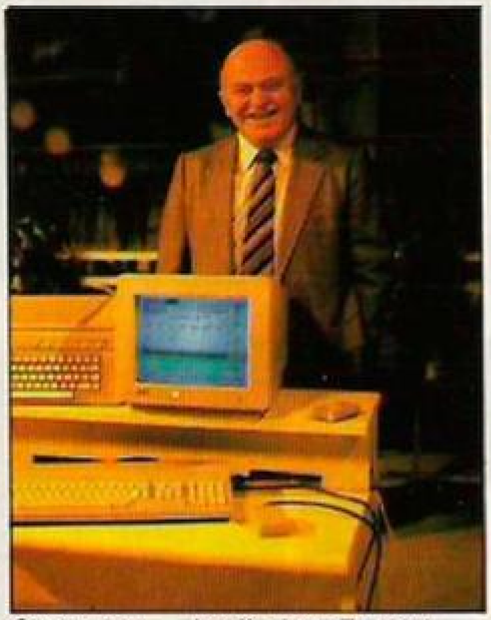
Crossfire for Spectrum and Amstrad

BBC Grand Prix and Dragon Spix

GRAFIX SPECIAL SPECTRUM, CBM-64, BBC AND ZX-81

YOUR COMPUTER Contents

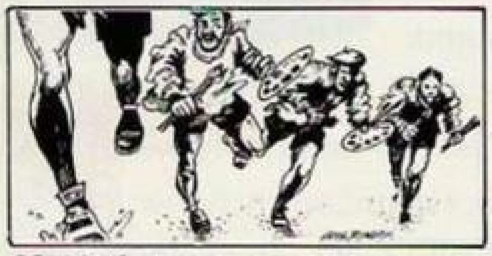
APRIL 1985



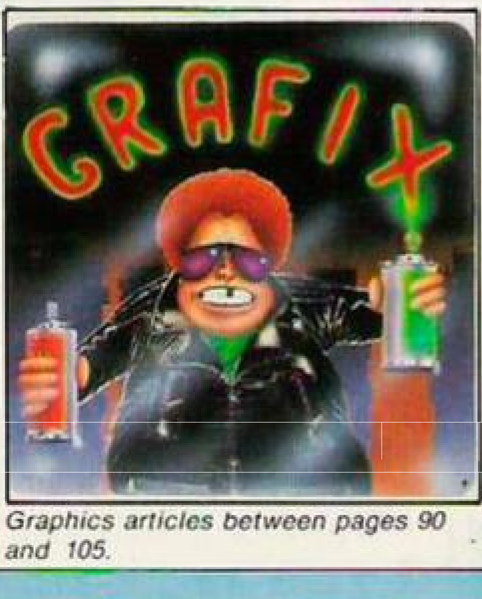
Cover shot — Atari's Jack Tramiel: page 54.



BBC Grand Prix Racer: page 62.



CBM-64 Quickplot: page 104.



Graphics articles between pages 90 and 105.

GOLD
IF YOU
YOU H

It's a range of top quality software from Commodore designed to make the most of your Commodore 64's capabilities.

New packages will be introduced to the Gold Medallion range every so often, but only if they are really exceptional. They'll definitely be hard soft to beat.

Miss any of them, and you really will be missing out.

MUSIC MAKER

Whether you're an accomplished musician or an out-and-out beginner, Music Maker strikes exactly the right note.

No matter if you've never played a note before, so long as you can hum and you know your ABC, you can start to play famous popular tunes immediately.



And it won't be long before you appreciate Music Maker's many advanced capabilities: you can synthesise many musical instruments, even create your own 'synth' electronic sounds, choose between monophonic or polyphonic play, summon up pre-programmed rhythms and bass accompaniments, and more.

Music Maker is the first in a series of packages which will fully exploit the Commodore 64's outstanding musical capabilities.

On disk or cassette, with music keyboard, a clear and concise manual, and song book, for just £29.95.

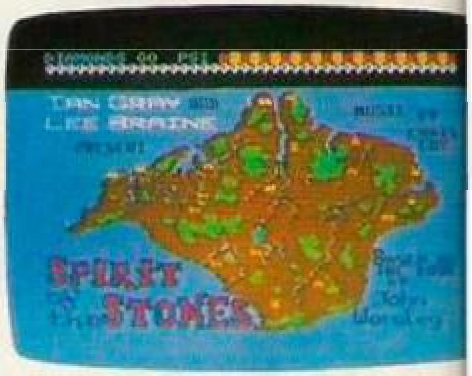
SPIRIT OF THE STONES

It's a treasure hunt for 41 real diamonds hidden somewhere on the Isle of Wight.

Only one man knows where they are, and he's not saying.

All that he has to say he's said already – but in the form of riddles, clues and puzzles.

Solve the puzzles in the Spirit of the Stones program and you'll find it much easier to solve the puzzles in the Spirit of the Stones book.



Solve the puzzles in the book and one (or more) of the diamonds can be yours.



Whoever discovers a diamond can also claim his or her share of the Royalty Fund, which could grow to a maximum of £1 million. It's a game that's as entertaining as it can be rewarding.

On cassette or disk, £14.99.

INTERNATIONAL FOOTBALL

Already it's recognised as the best football game ever seen outside of Wembley.

It's startlingly life-like, and gives you near perfect control of the players.

You can kick the ball, dribble it, pass it,

FOR MORE INFORMATION CONTACT: COMMODORE INFORMATION CENTRE, 1 HUNTERS ROAD, WELDON, CORBY, NORTHAMPTON NN7 1QX. TEL: CORBY (0536) 2062

At 21.15 hours on the night of May 16th 1943 a flight of specially prepared Lancaster Bombers led by Wing Commander Guy Gibson of 617 Squadron left Scampton Airfield for a mission so daring, that if successful, would do immense damage to the German war machine and change the course of World War II.



'Dambusters' Lancaster Bomber reproduced by kind permission of the Royal Air Force Museum, Hendon.

On May 16th 1985, a computer program will be launched, so full of excitement and realism that it will change the course of games software production.

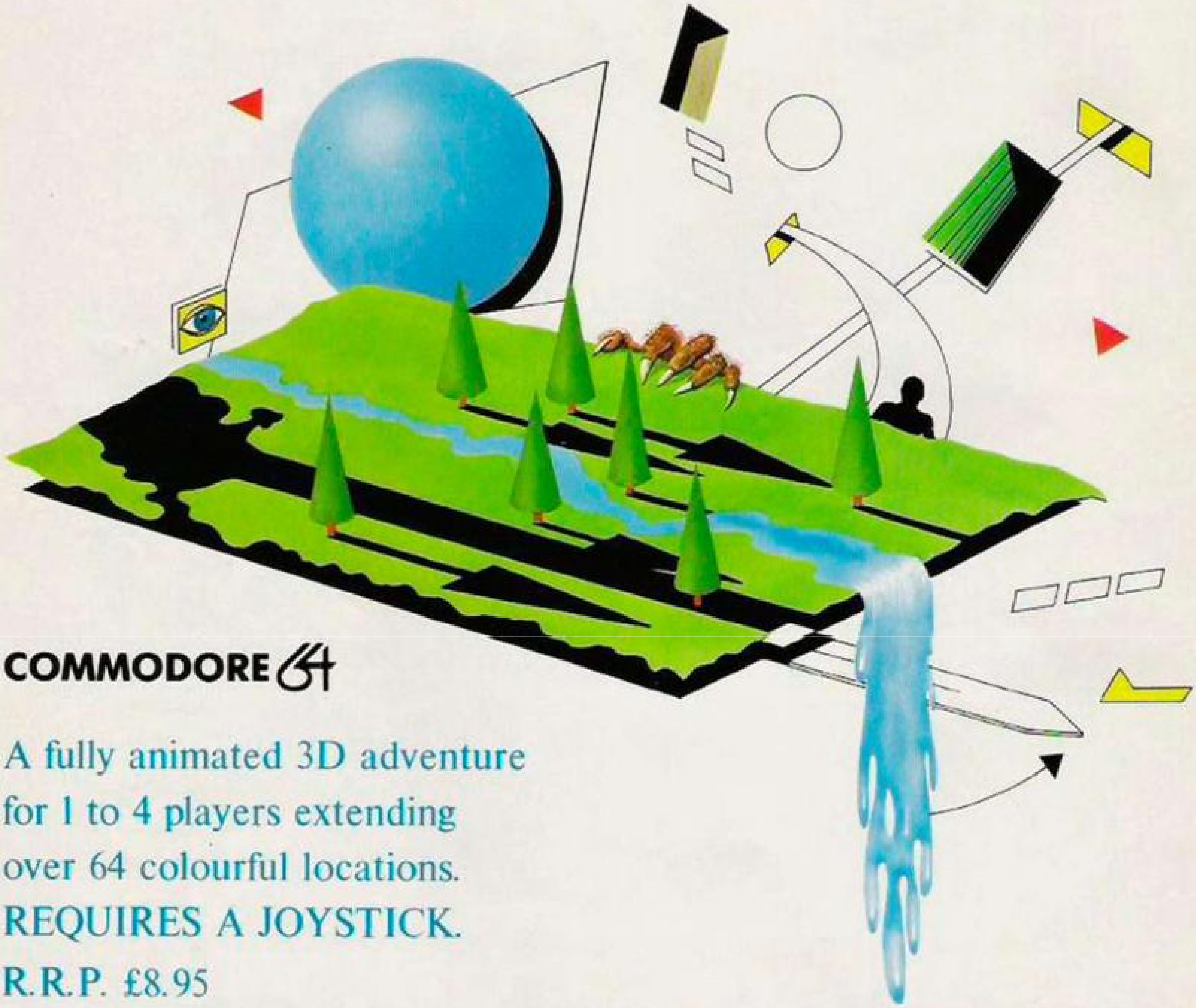
THE DAM BUSTERS

You will relive the spine chilling experience of this famous raid as you take the place of some of the bravest men in recent history. You will play the role of pilot, navigator, gunner and bomb-aimer in this real-time multiscreen, arcade/strategy game which is an authentic recreation of that fateful flight.


3D
Vortex
SOFTWARE
CYCLONE
SINCLAIR SPECTRUM 48K
3D
Vortex
SOFTWARE
TORNADO
LOW LEVEL
SINCLAIR SPECTRUM 48K
JOYSTICK COMPATIBLE
VORTEX.
TAKING THE SPECTRUM
WORLD BY STORM.



A prophecy is about to be fulfilled. The Dead will rise again to eat the flesh of the living...
Zombie Zombie
From the authors of ANT ATTACK, Sandy White and Angela.
DJR
Softsolid 3D* from SPACEMAN
*Patent pending
48K Spectrum £6.95
All titles available from
QUICKSILVA
Quicksilva Mail Order, P.O. Box 6, Wimborne, Dorset BA21 7PY.
Telephone (0202) 891744.
WARNING: These programs are sold according to QUICKSILVA Ltd's terms of trade and conditions of sale, copies of which are available on request.
WHSMITH
Boots
WOOLWORTH
John Menzies
and leading multiple retailers and specialist computer stores.





QL NEWS

News from the world of Sinclair QL computing.



The communications explosion takes shape!

Communications are now the most exciting, essential part of any computer.

In the past six months alone, over 150,000 modems have been sold in the UK.

Now, the QL's own communications explosion is taking shape... and it has the potential to make more of communications than any other micro!

Read on and discover exciting new ways to use your QL... with the QL modem... telephony unit... and powerful interface options.

No 2



DAVID KARLIN

Why Q COM is everything you could wish for in communications.

The QL is now communicating – via Q COM! This exciting three-part peripheral presents QL users with a multitude of ways to exploit the world of communications.

Once connected to the QL, Q COM allows you to access the considerable number of phone-in databases, such as Prestel and QNet.

Q COM enables you to communicate with other computer users. Its facilities include electronic mail, data transfer from Microdrives and bulletin boards.

Through it you can link your QL to larger minicomputers. Q COM has full capability in this area, and allows the QL to talk to powerful mainframes.

Q COM's automatic dialling and call acceptance facilities, together with the storage of messages from other modems, will revolutionise the way you use your telephone.

The next few pages of QL News tell you much more.

It's enough for me to say here that with the QL and Q COM, you'll be exploring new openings in communications for some time to come!

David Karlin

David Karlin,
Chief Design Engineer.



The Q COM package

Three special parts to stack!

QL communications interface

Q CON

This multi-speed interface contains the sophisticated software used to set up QL communications – and to control the Q CALL and Q MOD units.

Q CON also comes complete with Microdrive-based software. This enables the QL to link to larger computers using VT100 and viewdata protocols.

The software will also run *any* standard modem – connected via Q CON's built-in RS-232-C port.

Most importantly, Q CON allows the QL to transmit and receive at rates switchable from 75 to 9600 baud (encompassing the widely-used 75/1200 Prestel rates, and 1200/1200 half duplex rates for user-to-user exchange).

Q CON is specially styled to suit your QL – with similar fluting and ribs – and forms the base module of a vertical-stacking system.

It's supplied with full instructions, software on Microdrive cartridge, and connecting leads.

QL auto dial/answer unit

Q CALL

Q CALL gives every QL user something out of the ordinary.

It's a module which links directly to your telephone, and allows auto-dialling at the push of a single key. In the same way, it will permit incoming calls to be accepted automatically . . . and even trigger pre-programmed activity from the QL!

Q CALL is the central unit of the package. It plugs directly into Q CON – so there are no connecting cables to worry about.

QL modem

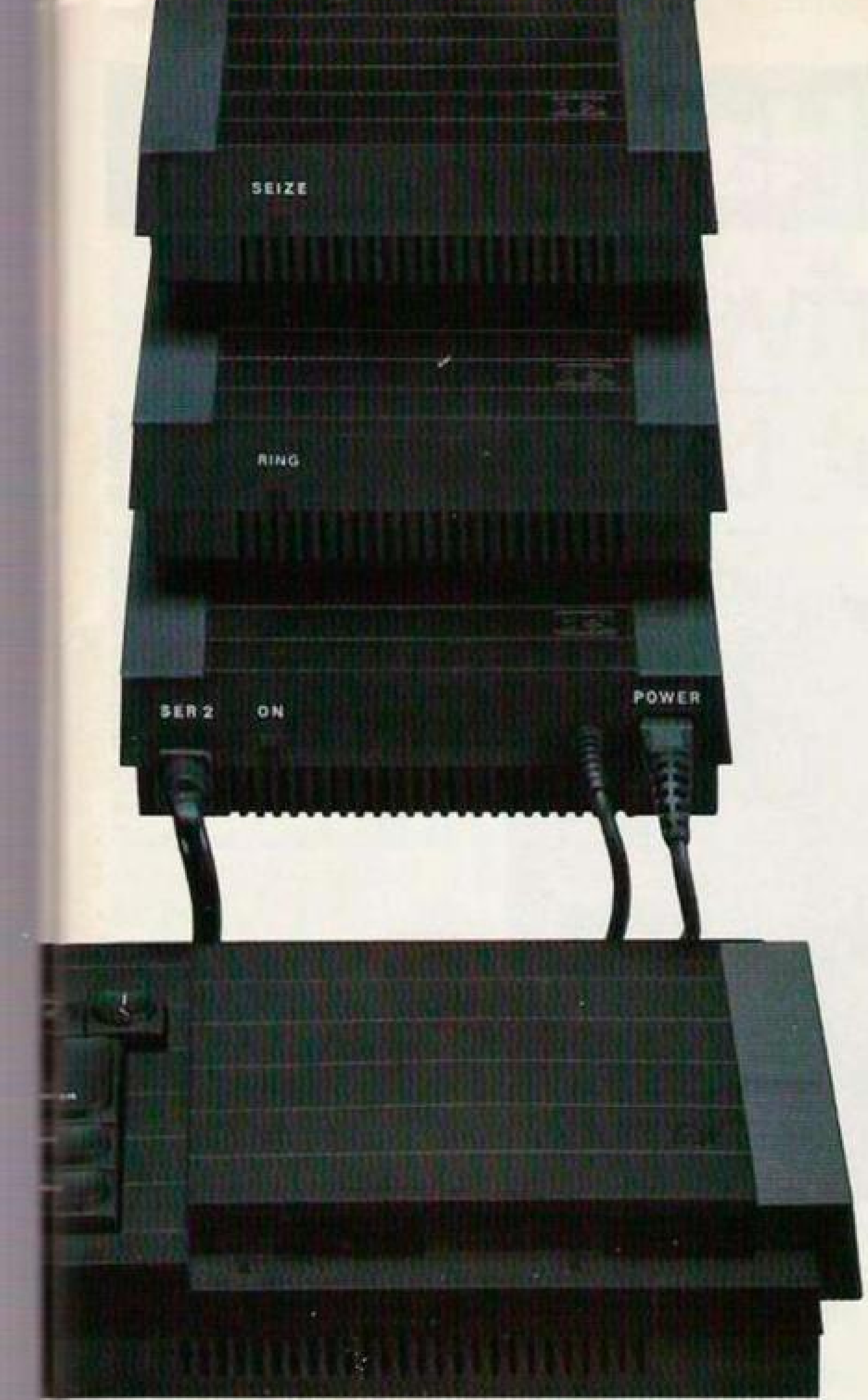
Q MOD

Q MOD has all the powerful facilities expected of a modem, in a neat and simple unit.

It uses either V23 75/1200 or 1200/1200 baud rates, for Prestel, Micronet 800 and all the viewdata services described alongside.

It also incorporates a telephone extension socket for manual dialling.

Q MOD is the top unit of



Q COM, and comes with a 9' built-in telephone cable.

All three units are available from OEL on (0276) 66748 and from selected Sinclair stockists.

The QL hooked on voice and data

The QL can now act as your personal address book and telephone operator!

Q COM allows you to store hundreds of personal or business numbers.

You can store lengthy passwords and account numbers – and recall them – at the touch of a single key.

And any information that's sent to you from other modem-owners can be gathered and stored on Microdrive cartridge, or incorporated into your QL Quill documents!



Exploring the world of QNet, Prestel, Micronet and more!

Thousands of QL users already enjoy the excitement of linking to a nationwide mainframe.

Q COM turns your QL into an intelligent terminal, allowing you to access many thousands of pages of information, software and communications facilities.

The services brought to you through Prestel can include Micronet 800, Viewfax 258 and QNet, the new QL database.

Membership of QNet will bring you free software, QL news and features, and all the wide-ranging services of viewdata!

If armchair shopping is more your style, that's easy too. It's often possible to place a direct order using your QL! For dedicated QL owners, there's a daily selection of software reviews, chart toppers … and all the facts and figures you need to make buying peripherals simple.

With Q COM you can also 'download' software from the system directly into your QL and either use it immediately, or store it on Microdrive cartridge.

In fact the only problem you'll face with a viewdata service is finding enough time to explore its many features!

You can find out how to join QNet by phoning 01-278 3143.



News … information … banking services and QNet. And only a fraction of the QL's new viewdata capability.

QL meets the mainframes!

The Q CON unit of Q COM turns your QL into a VT100 terminal, providing instant access to in-house computing services, both mainframe and mini.

Whether you are using your QL at home or at work, Q COM gives you access to electronic bulletin boards which provide help and advice 24 hours a day. You can leave messages or notices for friends or business contacts and even hold live discussions with them.

Additional benefits for the QL business user include easy access of in-house company software, and the interrogation of other data bases around the country.

There's also the opportunity of linking to British Telecom Gold – the widely-publicised and popular messaging service.

sinclair

QL Hardware

Microdrive cartridge price cut to only £1.99!



Sinclair Microdrive cartridges – up to 100K of programs and data on a medium so compact you can pop it into your pocket.

On February 1, the cost of Microdrive cartridges came down from £4.95 to £1.99 each.

Microdrive cartridges are the QL's own unique storage medium. Each stores up to 100K of information (that's 40 pages of A4 text), on a cartridge no bigger than a book of matches!

Over 500,000 cartridges are now being used throughout Britain.

You can store up to 50 different data files per cartridge, identified by titles of your own choice.

And QL Microdrives themselves are standard equipment on the new ICL One Per Desk micro, and British Telecom's new Merlin Tonto.

IEEE-488... the instrument connection

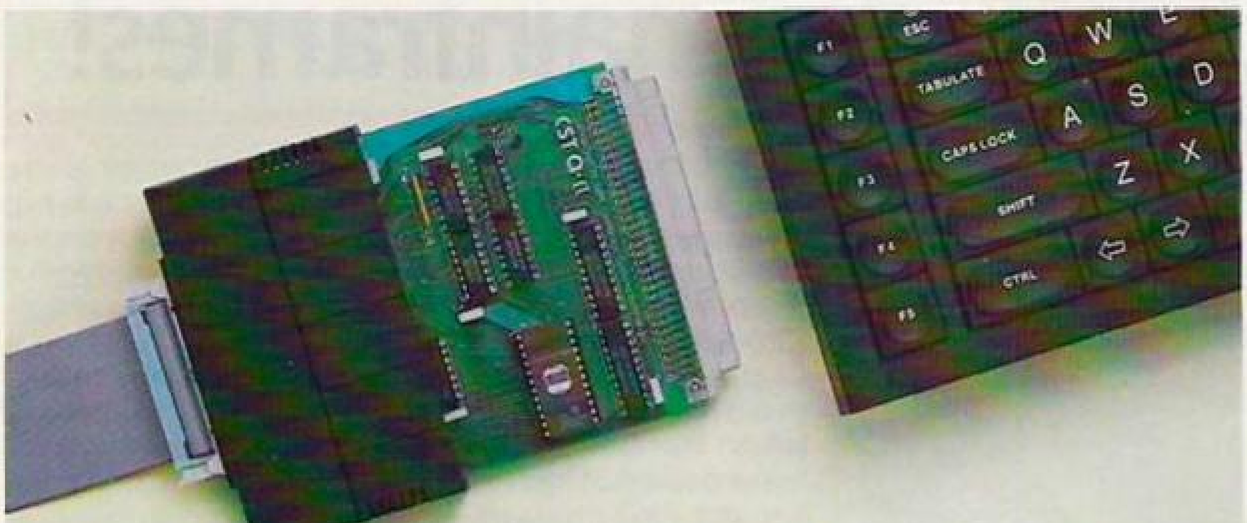
IEEE-488 is the interface standard set by the Institute of Electronic and Electrical Engineers for instrumentation control.

IEEE-488 – or General Purpose Instrumentation Bus – is a parallel interface specifically designed for high speed data transfer between a number of different types of device.

It is commonly used for controlling instrumentation via a computer, allowing the creation of laboratory data acquisition systems, industrial control schemes, etc.

The QL now has a fully-fledged IEEE-488 interface from CST. It plugs neatly into the QL's RAM expansion port, and can control up to 16 instruments simultaneously.

It's available from CST on (0223) 323302.



An IEEE-488 interface slips discreetly into place.

New inte 3½" or 5

With new Q-Disk, you can transform the QL into a powerful small business system – comprising QL, monitor, disk interface, twin disk drives and printer.

Q-Disk upgrades the QL to disk storage. Fitting easily into the QL's left hand RAM expansion port, without the need for a special expansion box, it contains a Western Digital disk controller chip. Software is held in an on-board EPROM (so little of the QL's RAM is used).

Plug in Q-Disk, and the QL accepts one or two disk drives, sized 3 in, 3½ in, 5¼ in, either 40 or 80 track, single or double-sided. Even when two drives are used, they can be different types!

Q-Disk offers up to 1.6 Mbytes of quick, reliable storage with a compatible disk drive.

It's made by Computamate, who also offer a full range of



QL to link students

Strathclyde University, in Glasgow, plans to have a campus network of 7,000 QLs linked to a central VAX minicomputer.

That's one QL for every student . . . a major investment project in a university which is now a leading centre for artificial intelligence work.

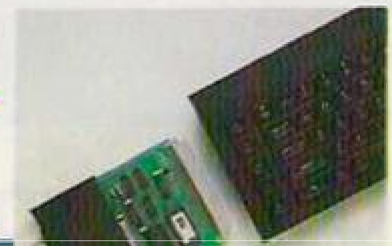
Sinclair is giving support worth £250,000 to the project. And it's likely that QL users everywhere will benefit – the students plan to develop AI programs to run on the QL!

The QL has impressed Prof. James Alty of the University's Computer Science Department, who says *'only the QL could offer the computing power, range of applications, and above all the portability, at a realistic price.'*

face to connect 3" disk drives

complimentary QL disk drives.

To contact Computamate, phone (0768) 811711.





Single disk unit fitted with 5¼ inch drives and *(inset)* the Q-Disk controller.

The QL's high-tech spec

Dimensions
138 x 46 x 472mm
(5⅜" x 1¾" x 18¾")

Weight
1388 gms (3.055 lbs)

RAM
Massive 128K standard RAM, externally expandable to 640K. Extra RAM is available in 64K, 128K, 256K and 512K units, from third-party suppliers.

ROM
48K, containing Sinclair SuperBASIC and the Sinclair Qdos operating system.

CPU
Motorola 68008 (running at 7.5 MHz) for all principal functions. (Architecturally, the 68008 is a 32-bit processor with an eight-bit data bus. One megabyte of non-segmented address space is available.)

In addition, an Intel 8049 controls the keyboard, generates the sound, and acts as an RS-232-C receiver.

Operating system
Qdos (developed by Sinclair Research) is a single-user multitasking, time-sliced system using Sinclair SuperBASIC as a command language with display handling for multiple screen windows; and device-independent input-output.

Language
Sinclair SuperBASIC, with the advantages of procedure structuring; extendability (including syntax); interpretation speed independent of program size; clean machine code interface; operating system facilities accessible from SuperBASIC; equal capability for strings and arrays; and full error-handling facilities.

Microdrives
The QL incorporates twin QL Microdrives, each with a minimum 100K capacity, 3.5 seconds average access time. Typical loading rate of machine code programs is 2-3K per second.

Video
High resolution graphics capability with colour or monochrome monitor (or TV) in two modes – 512 x 256 pixels (four colours available) and 256 x 256 pixels (eight colours available). Normal character display format of up to 85 x 25 with choice of character sets available (TV format of up to 40 to 60 columns depending on the software).

Keyboard
Full-size, 65-key QWERTY keyboard featuring a space bar, left- and right-hand shift keys, five function keys and four cursor control keys. The keyboard can be angled by means of detachable feet.

Expansion
Excluding RGB monitor, power socket and TV port, eight peripheral/expansion ports are provided – one internal expansion, one Microdrive expansion, one ROM cartridge, two serial and two control channels, and the local area network.

Serial
Two standard RS-232-C communications interfaces for printers, modems, etc. Transmission at rates from 75-19200 baud or full duplex transmit/receive at seven rates up to 9600 baud.

LAN
For up to 64 QL computers. Data transmission over the net can be achieved at 100K baud.

Power supply
9V DC at 1.8A, 15.6V AC at 0.2A.

Joysticks
Provision for one or two devices for games or cursor control.

Applications Software
QL Quill – word processor
QL Abacus – spreadsheet
QL Easel – graphics
QL Archive – database
All four packages supplied with the QL.

Price
£399 including VAT, QL programs, full A4 manual, power supply, 4 blank cartridges and free Helpline service.

sinclair

QL Software

Updated versions of Psion software now available!

QL Abacus, Archive, Easel and Quill are the four Psion programs supplied with every QL. They're now converted to 100% machine code, and as a result they load from Microdrive cartridge much faster.

The overlays present in Version One software have been removed, resulting in noticably quicker on-screen performance.

With the compactness of machine code, there's a big saving in QL memory too – all four programs now cope with larger, more professional applications!

Version Two software is now supplied with every new QL. Existing QLUB members – see back page.

QL·Quill

QL Quill makes it easy to type in, correct and store your letters, memos and reports.

No training is needed – a beginner can be using QL Quill for word-processing within minutes!

QL Quill has the facilities of professional word processing packages: including word wrap, search and replace, justification, page headers and footers.



QL·Abacus

QL Abacus is a powerful, yet easy-to-use spreadsheet.

The program allows you to manipulate the contents of whole rows and columns by the names you assign them. There's no need to depend on confusing letters and numbers.

QL Abacus also incorporates a range of functions which let you carry out rapid 'what if' analyses on your data.



QL·Easel

QL Easel allows you to create graphs, bar charts and pie charts – at the touch of a key.

The program handles anything from lines and shaded curves to overlapping or stacked bars.

QL Easel designs and scales automatically or under your control. Text can be added and altered as simply as data.



QL·Archive

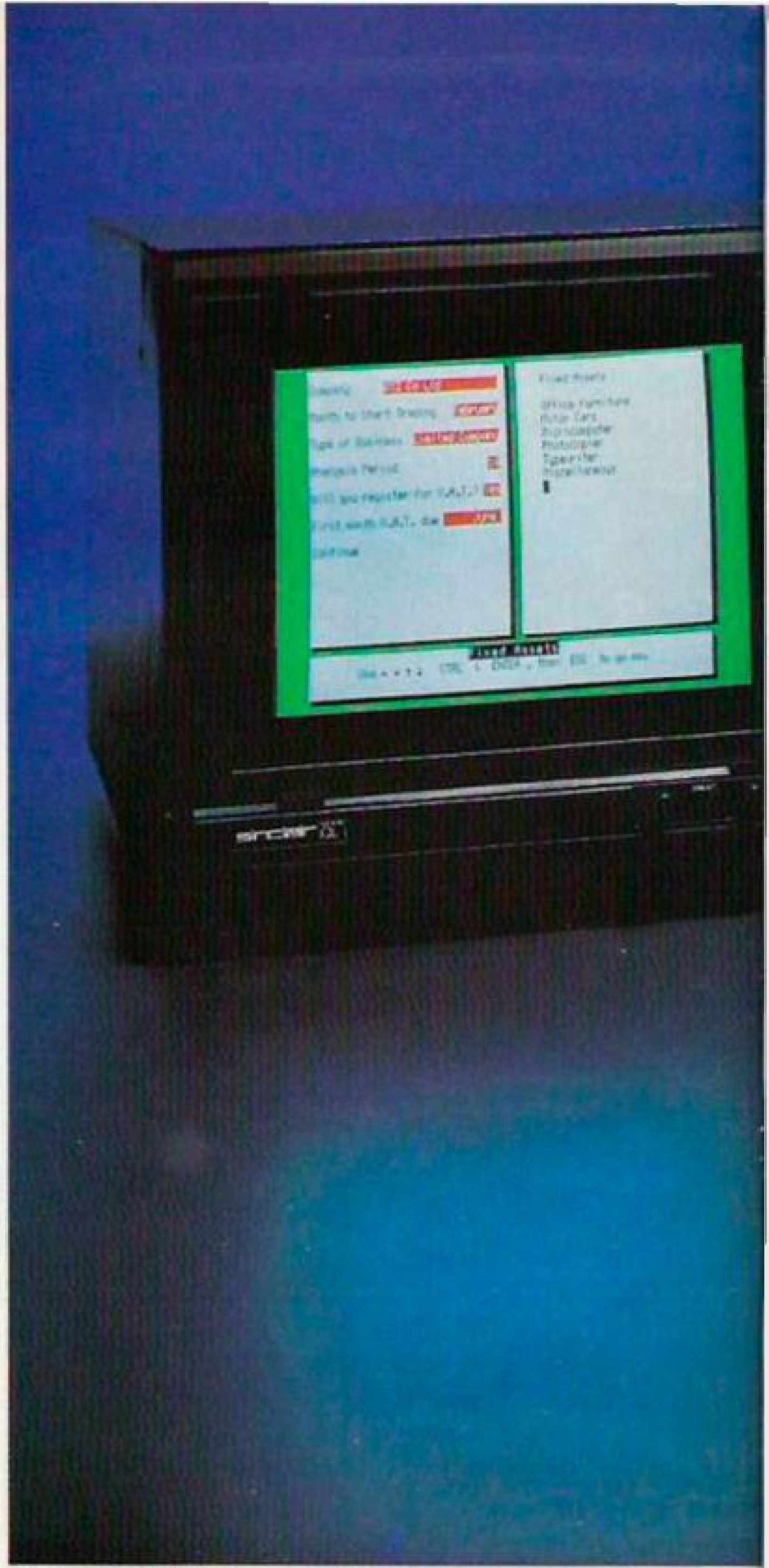
QL Archive is a sophisticated, powerful database program.

It includes a screen editor which allows you to design your own screen and format your reports, and a procedure editor which lets you tailor QL Archive to your own requirements.

QL Archive is ideal for all database uses, yet it's powerful enough to be used by many software houses to generate specific database applications.



Non-members of QLUB can purchase new versions of the above software for £15 per title, or £50 for all four programs. Phone (0276) 686100 for details.



(Left to right) QL Entrepreneur, QL Project Planner and QL Decision Maker from Sinclair.

Coming soon- QL·Entrepreneur, QL·Project Planner, QL·Decision Maker!

Three new QL business programs – with a difference!

QL Entrepreneur, QL Project Planner and QL Decision Maker train you to apply new and exciting management skills – through original and powerful means!

An interactive teaching program gives you a thorough and enjoyable understanding of each subject – backed by a text-book and 'self-test' facilities - and an applications program helps you to use your new expertise for specific problems and projects.

All three titles will increase your understanding and extend your control – making involved subjects easy, stimulating and useful!



QL·Entrepreneur

QL Entrepreneur is an essential program for anyone preparing to start a new business – whatever it may be!

It uses a 'question and answer' format to help you build a workable business plan.

With the input you give, it works out the break-even point of the business; the first 18 months' cash flow, the type of finance needed; the year end Balance Sheet and Profit and Loss accounts . . . and more!

QL Entrepreneur builds your skills and techniques.

It's flexible too, so that you can ask complex 'what if' questions at any stage!

The program comes with a third, blank Microdrive cartridge and a comprehensive A5 manual.

QL·Project Planner

QL Project Planner will produce plans you can understand, monitor and more easily achieve.

First, you break the project down into its individual activities, telling QL Project Planner how long each takes and which are inter-dependent.

When you decide on a starting time/date QL Project Planner will tell you when each activity must start and finish and when the project will be completed.

Each activity is divided into its critically important stages – those which can safely be moved around without altering the time taken by the project and those where movement will affect the completion deadline.

Whether or not you've used project planning systems before, you'll be amazed at the difference QL Project Planner can make.

The program comes with a third, blank Microdrive cartridge and a comprehensive A5 manual.

QL·Decision Maker

Whether you're thinking of buying a house, or taking on a new business contract, QL Decision Maker makes the choices clearer!

It lets you look at the possibilities – and their implications – through a decision tree.

Once you've set out the decisions and their probable costs or results, QL Decision Maker shows the outcomes which would occur from each particular route.

You can see how much money a decision could make for you . . . or cost you. Complex 'what if' questions are dealt with swiftly and graphically.

You can depend on the QL to highlight the best possible route!

QL Decision Maker comes with a third, blank Microdrive cartridge and a comprehensive A5 manual.

All three programs are available from Sinclair stockists, price £39.95 each, or Sinclair Research. Tel: (0276) 686100.

sinclair

Now, buy a QL and you're a member of the QLUB-free!

QLUB is the special Users Bureau for Sinclair QL owners.

Already, there are well over 10,000 QLUB members . . . enjoying a whole range of information and advisory services.

Until now, joining QLUB cost £35 per year. From March 4, every new QL owner can become a member – free of charge!

With your new QL, you'll find a post-paid form. Complete and mail it, and you'll soon be a member of the fastest growing computer club in the country.

And you'll enjoy all the helpful services listed here!

What QLUB membership offers you

Regular newsletters delivered to your door

One of the most important QLUB benefits is the special news magazine, appearing six times a year. The magazine provides a forum for QL owners to exchange views and keep in touch with all the latest developments.

Each issue is packed with updates on QL hardware and software, tips on applying the four QL programs, and news of how other people are using the QL.



Special discounts

QLUB members also receive a range of special discounts, with savings of at least 20% on selected software products.

There are also special subscription rates for Personal Computer News and QL User.

Free Helpline service from Psion

All QLUB members are entitled to 12 months special assistance from Psion.

They're at the end of the telephone to answer any questions on using the QL Abacus, Archive, Easel and Quill programs supplied with the computer.

Help is also available on any aspect of using Sinclair SuperBASIC, Qdos, or linking your QL with major peripherals.

Psion will normally answer any queries within 48 hours.

QL program updates are no longer available free to QLUB members. They will be sold separately.

Good news for existing QLUB members too!

As one of the first members of QLUB, you should already have received one *free* update of each of the four QL programs – and a letter with your new membership details.

If for any reason you haven't, you should ring (0276) 686100.

You're a QL owner, but not a QLUB member?

Then joining QLUB is easy and free! Ring (0276) 686100 for full details. You can be a full QLUB member within a few days.

Where to find the QL

The Sinclair QL is available at selected branches of Dixons, WH Smith, John Lewis Partnership, Currys, Greens in Debenhams and Ultimate, and larger branches of Boots, John Menzies and specialist computer stores nationwide.

sinclair, QL, QLUB, Qdos, and SuperBASIC, are Trade Marks of Sinclair Research Ltd. Quill, Easel, Archive and Abacus are Trade Marks of Psion Ltd.

Sinclair Research Ltd
Camberley, Surrey, GU15 3BR.
Tel: Camberley (0276) 686100

sinclair

YESTERDAY'S INNOVATION... TODAY'S STATE OF THE ART!



The Hobbit – still the most sophisticated and exciting adventure for your micro.

Available on cassette for the Spectrum, Commodore 64, BBC, and soon to be available for Amstrad and MSX computers at £14.95.

New release on disk at £17.95 for BBC – including graphics – and Commodore 64, with many new exciting features, including enhanced graphics and music.

Enquire at your local computer store, Boots, W.H. Smith or John Menzies.
Trade enquiries call Melbourne House on 01-940.6064.

NOW ON DISK.



Herbie Briggs has just destroyed the myth that all floppy discs are created equal.

They seem equal. Until you look at the seams.

That's where equality ends.

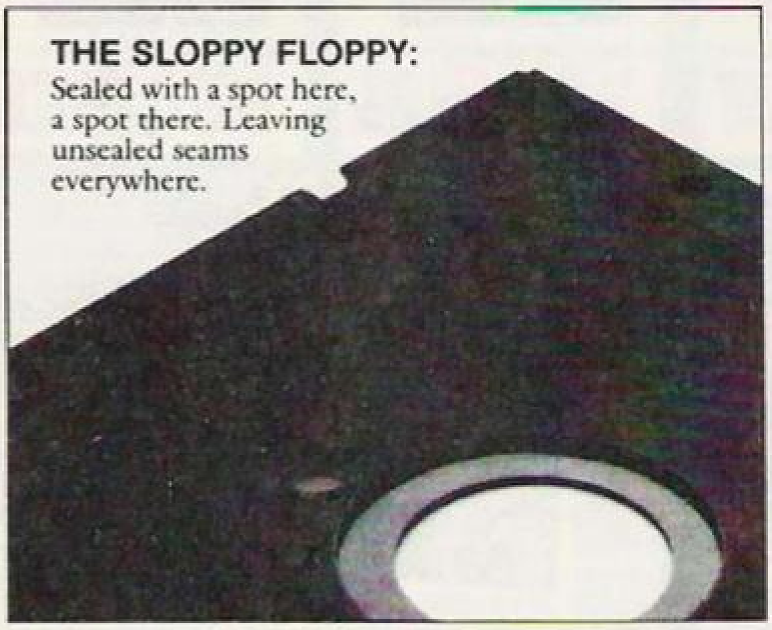
Most companies seal their discs with a spot here, a spot there. Leaving most of each seam not sealed at all.

Sooner or later, the seams might do what comes naturally: they bulge. Warp. Pucker. Open up.

Pens, pencils, fingernails—even a four-year-old's, like Herbie—can catch and snag in those wide open spaces.

That's sloppy. And dangerous. Because if you put a sloppy floppy into your disc drive, it can jam your drive. Ruin your drive head. Lose your data.

So much for their seams. Ours are different.



Memorex uses a process we developed, called Solid-Seam Bonding.

Solid-Seam Bonding seals shut every inch of every seam of every Memorex® floppy disc. Tight as a drum. That makes the Memorex floppy stiffer. Stronger. And your data safer.

To resist bulging, warping, puckering, or opening up.



To resist all the things that can jam your drive, ruin your drive head, or lose your data.

Which proves that a Memorex floppy disc isn't equal to all the others. It's better.

Solid-Seam Bonding is just one example of the extra care that goes into every Memorex floppy disc. Be it 8," 5¼" or the new 3½." Extra care that lets us guarantee every Memorex disc to be 100% error-free.

The next time you're buying a floppy disc—or a few hundred of them—just remember this:

It's always better to be safe than sloppy.

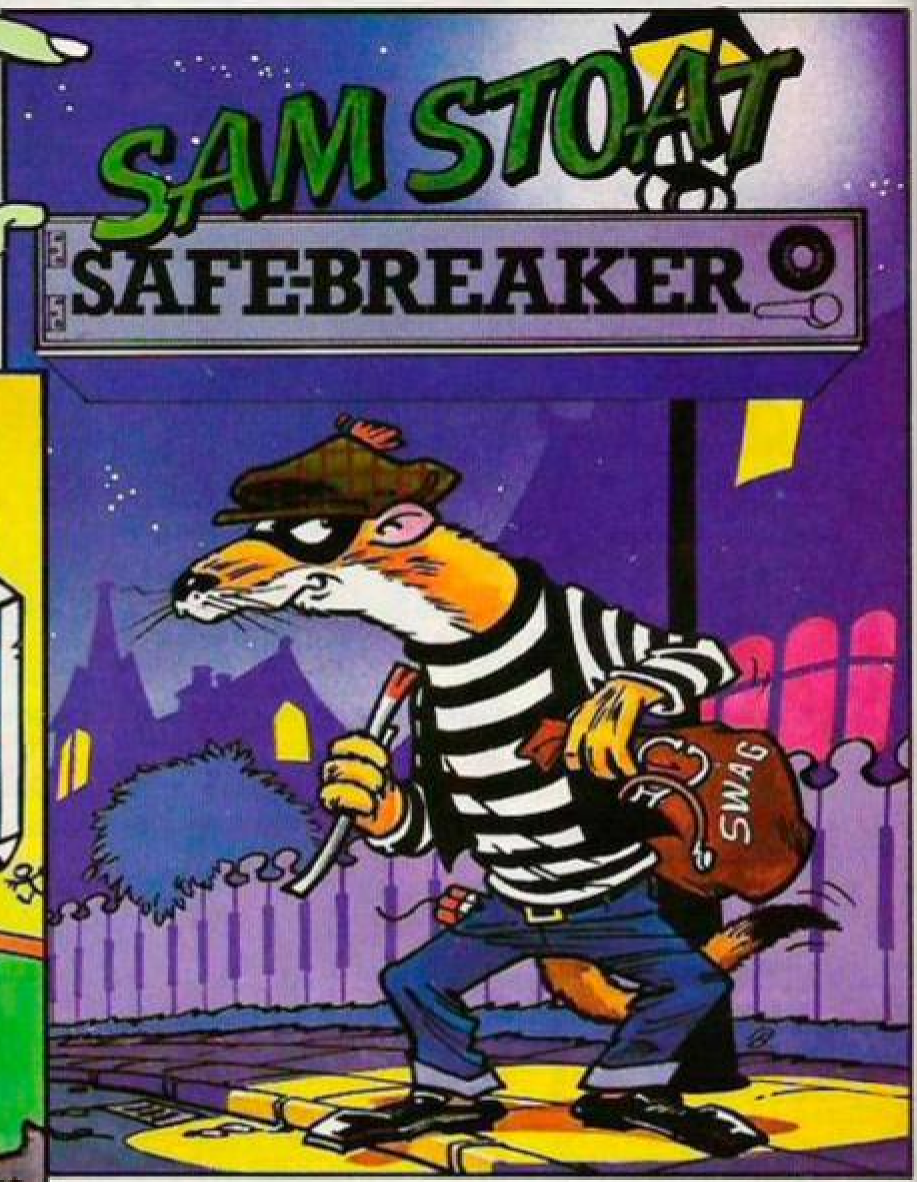
For more information on the full line of Memorex quality computer media products, including computer tape, call Memorex U.K. Ltd., 96-104 Church Street, Staines, Middlesex. Tel: 0784 51488

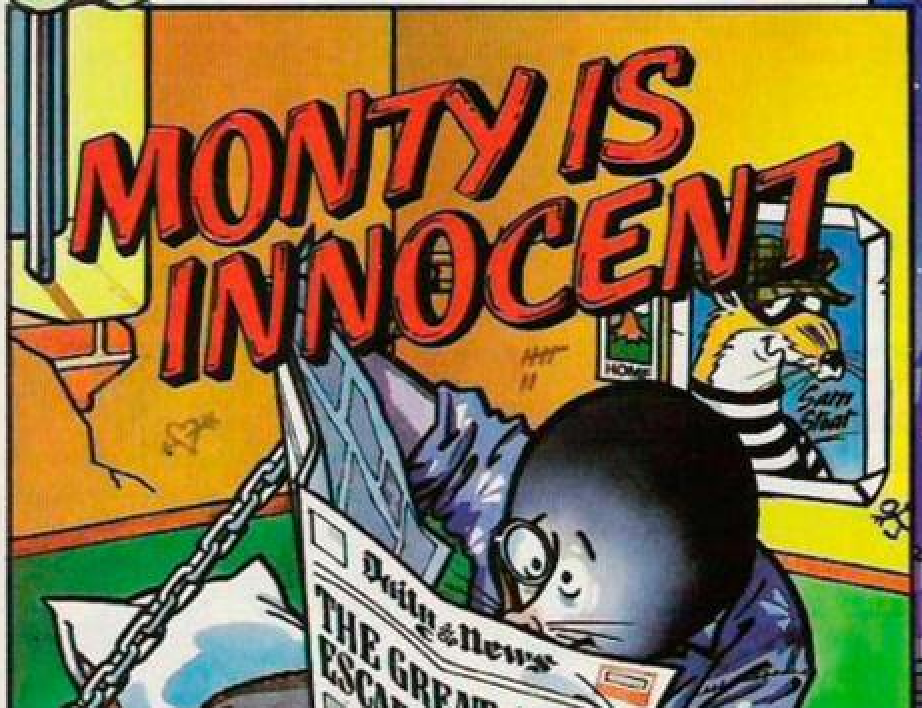


Your Data. When it matters, make it Memorex.™

MEMOREX







At the dead of night Sam leaves his lair and makes for the large houses where, with the aid of some everyday tools and a good measure of cunning, there are wealth and riches for his delight (not to mention his pocket). It couldn't be easier. . . sneak into the house, find the safe; light the blue touch paper and stand well back. Then out into the night with the ill-gotten gains! But Sam has reckoned without one or two adversaries who are more than a little determined to get the better of him. . .

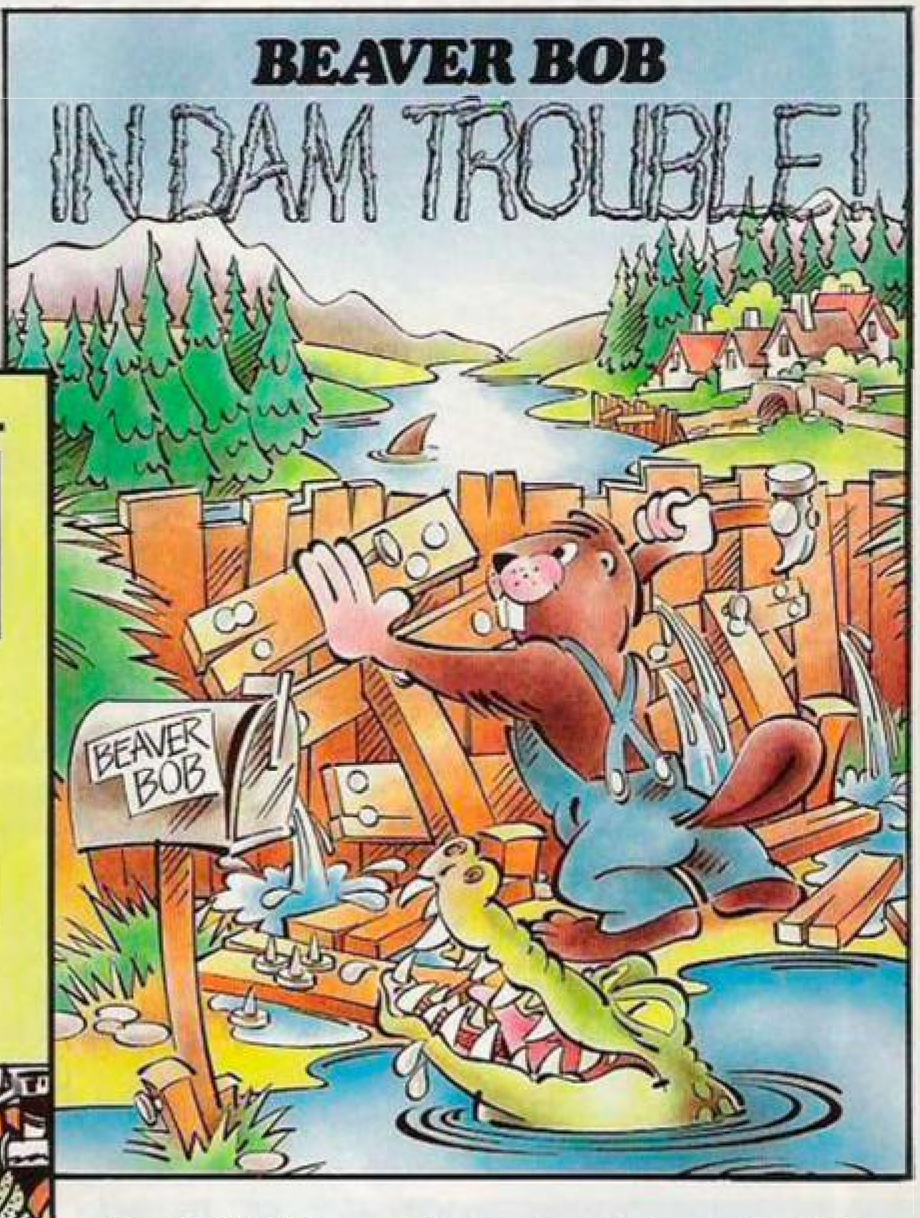
A cry of national outrage! Poor Monty banished for life amongst murderers, arsonists, the lowest of the low. Fear not, this Mole is innocent, and plans are in hand to rescue your Superhero. Who is the mysterious masked rodent willing to challenge death for his lifelong friend? What are the evil forces working in Scudmore Prison?

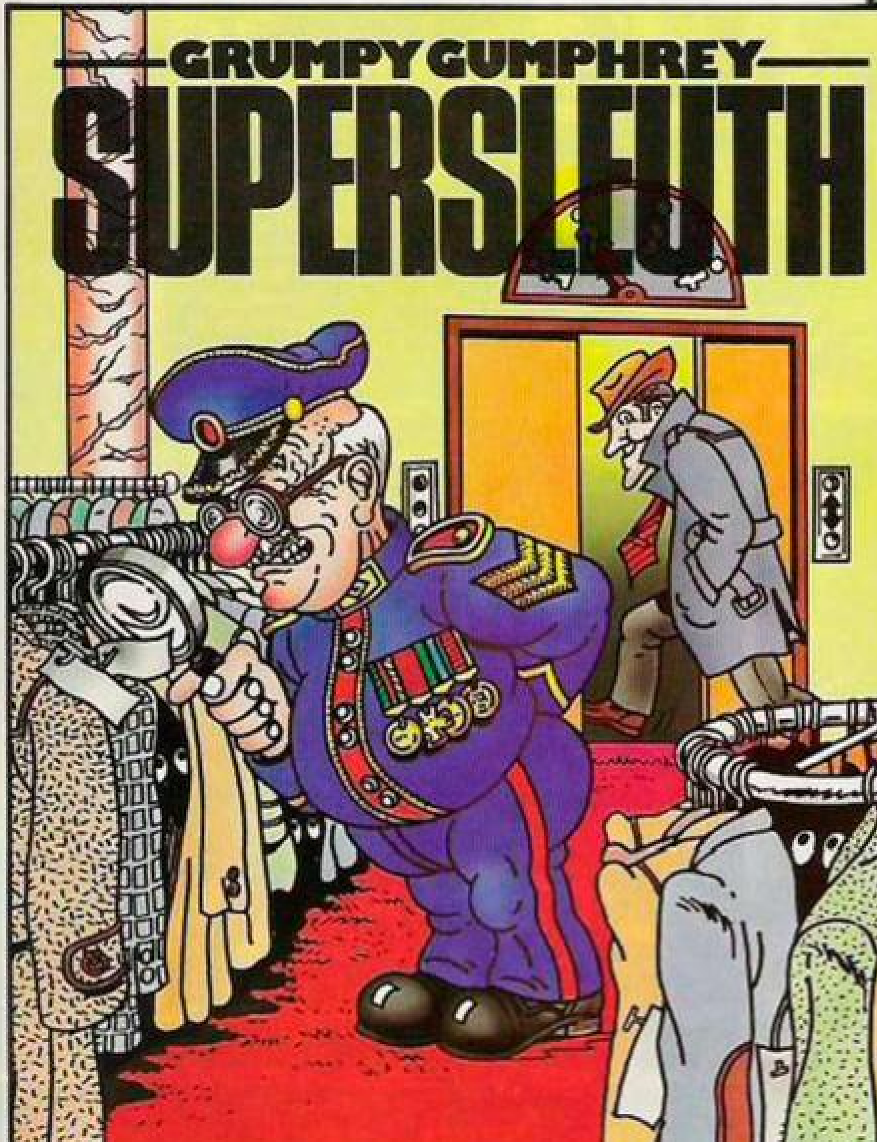
OTHER GREAT TITLES FROM GREMLIN:

Suicide Express	CBM 64 **£7.95**
Monty Mole	CBM 64 **£7.95**
	Spectrum 48k **£6.95**
Potty Pigeon	CBM 64 **£7.95**
	Spectrum 48k **£6.95**

OU'VE NOT GOT THE LINS, YOU DON'T KNOW AT YOU'RE MISSING!

Ex serviceman Gilbert Gumphrey, drummed out of the Army in 1946 because of his acute short-sightedness has been struggling to hold down his job as a store detective for the last thirty nine years. But things are hotting up for Gilbert, known to all as "Grumpy". Now that times are becoming harder and staff made redundant, he has to perform all kinds of tasks; not only catching thieves, but caging the gorilla and tidying the warehouse, otherwise he might find himself surplus to requirements too! Can Gilbert hang onto his job? You can help him, but be warned — this is no ordinary department store!





Poor Bob! All he wants is to lead a quiet life and build his dams. But the peace is soon shattered when the local townsfolk find their taps running dry. Bob's dams are plainly to blame so as soon as he completes one, the locals blow it up! And that isn't the end of Bob's troubles either, for there are sharks, crocodiles and all manner of underwater hazards for him to negotiate. Help Bob in his never ending struggle to build his dams and avoid the predators that lurk in the deep.

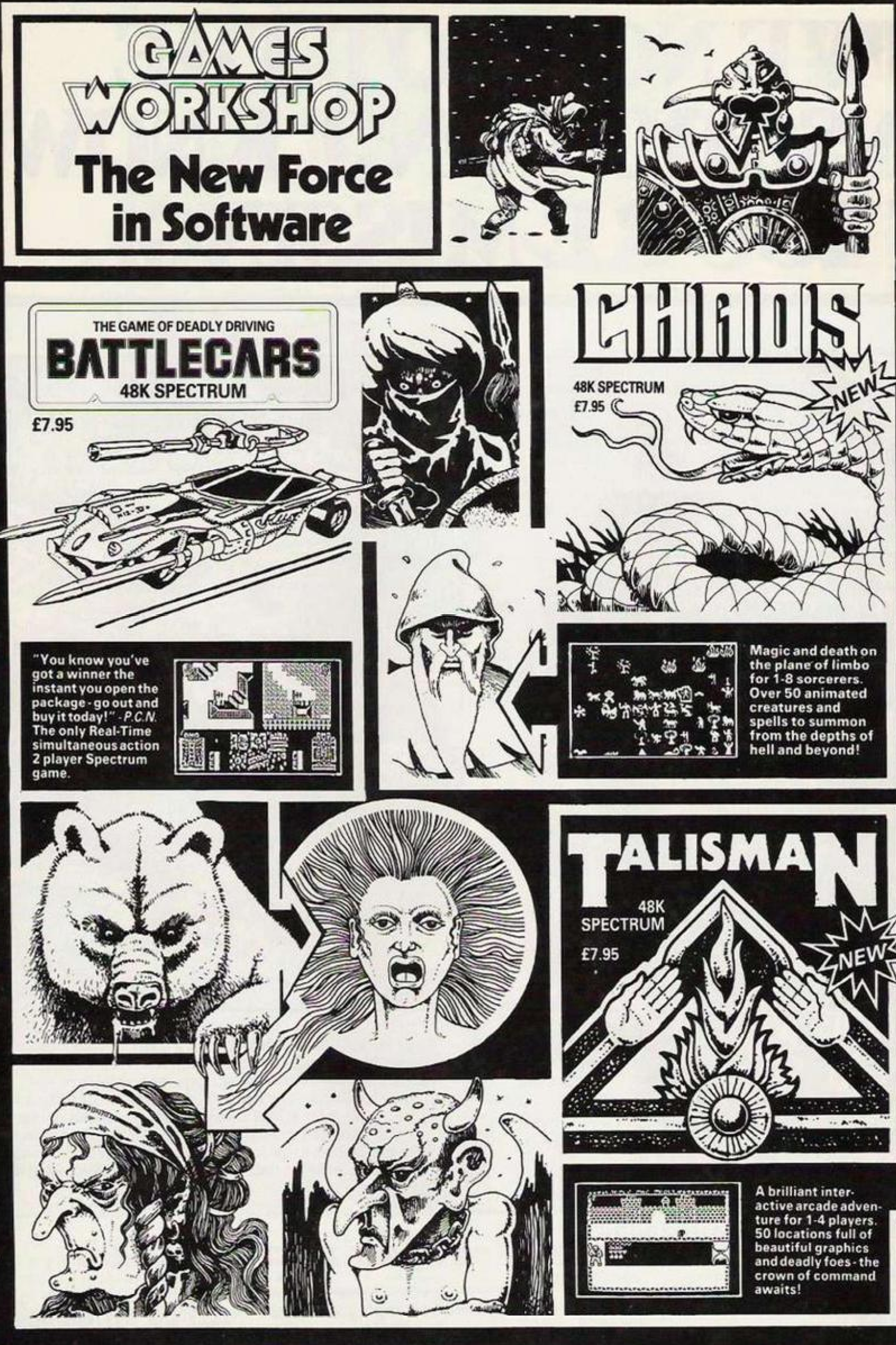
All titles available for Spectrum 48k £6.95

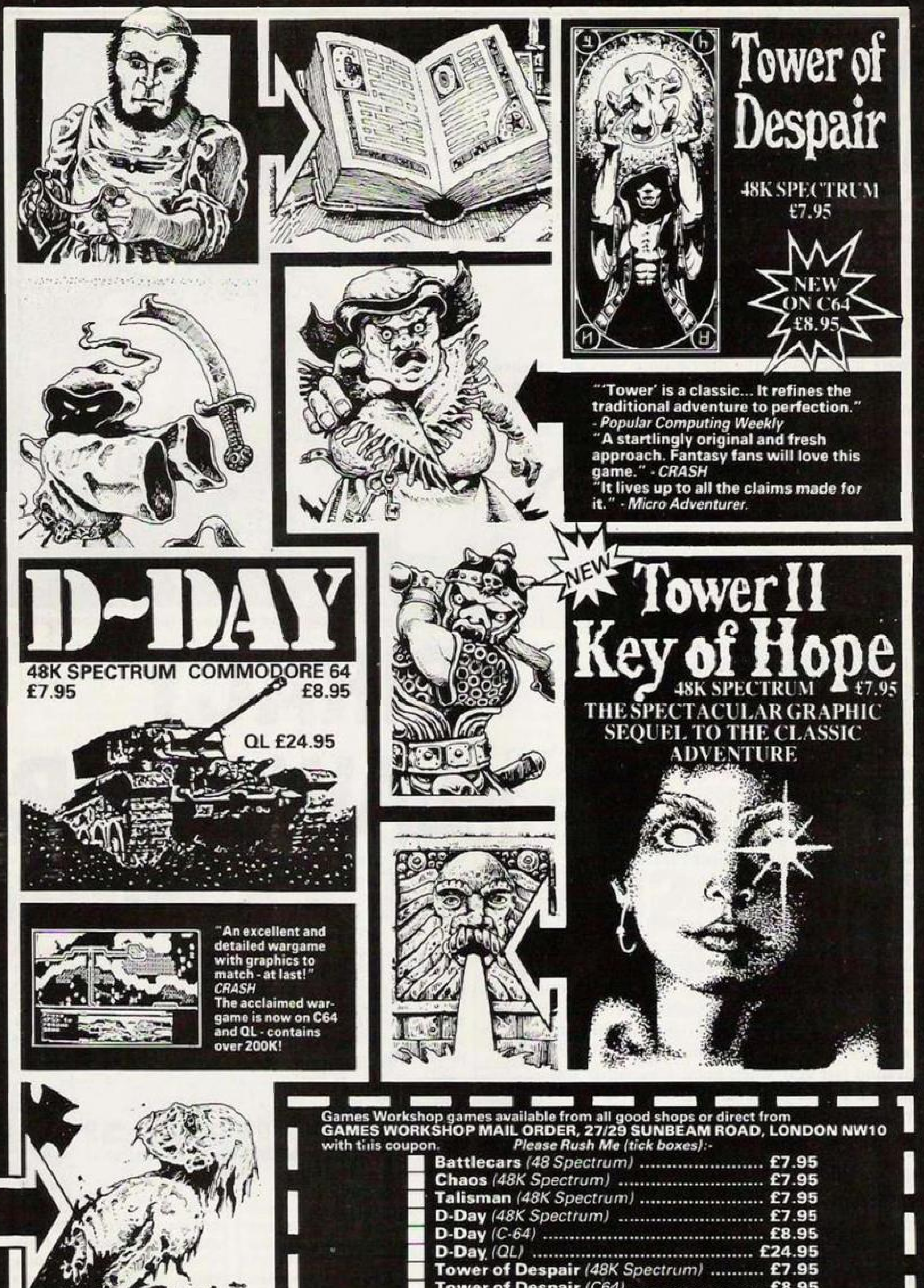
For amazing graphics and outstanding sound effects run a gremlin through your micro.

Gremlin Graphics, Alpha House, 10 Carver Street, Sheffield S1 4FS. Telephone: (0742) 753423

Gremlin Graphics

es
e
ne
o
95
95
95
95
95


GAMES WORKSHOP
The New Force in Software
THE GAME OF DEADLY DRIVING
BATTLECARS
48K SPECTRUM
£7.95
"You know you've got a winner the instant you open the package - go out and buy it today!" - P.C.N.
The only Real-Time simultaneous action 2 player Spectrum game.
CHAOS
48K SPECTRUM
£7.95
NEW
Magic and death on the plane of limbo for 1-8 sorcerers. Over 50 animated creatures and spells to summon from the depths of hell and beyond!
TALISMAN
48K SPECTRUM
£7.95
NEW
A brilliant interactive arcade adventure for 1-4 players. 50 locations full of beautiful graphics and deadly foes - the crown of command awaits!



Tower of Despair
48K SPECTRUM
£7.95
NEW ON C64 £8.95
"'Tower' is a classic... It refines the traditional adventure to perfection." - Popular Computing Weekly
"A startlingly original and fresh approach. Fantasy fans will love this game." - CRASH
"It lives up to all the claims made for it." - Micro Adventurer.
D~DAY
48K SPECTRUM COMMODORE 64
£7.95 £8.95
QL £24.95
NEW
Tower II
Key of Hope
48K SPECTRUM £7.95
THE SPECTACULAR GRAPHIC SEQUEL TO THE CLASSIC ADVENTURE
"An excellent and detailed wargame with graphics to match - at last!" CRASH
The acclaimed wargame is now on C64 and QL - contains over 200K!
Games Workshop games available from all good shops or direct from
GAMES WORKSHOP MAIL ORDER, 27/29 SUNBEAM ROAD, LONDON NW10
with this coupon. Please Rush Me (tick boxes):-
Battlecars (48 Spectrum) £7.95
Chaos (48K Spectrum) £7.95
Talisman (48K Spectrum) £7.95
D-Day (48K Spectrum) £7.95
D-Day (C-64) £8.95
D-Day (QL) £24.95
Tower of Despair (48K Spectrum) £7.95
Tower of Despair (C64) £8.95
Key of Hope (48K Spectrum) £7.95
Cheques/POs made payable to Games Workshop Ltd.
Post & Packing free in UK. Overseas add 15% to total cost.

WHY WHIZZ KIDS ARE GOING GREY

HALF THE SOFTWARE houses in Britain could be wiped out overnight if one more major distributor like Prism goes bust. That's what the prematurely-aged whizz-kids of 1982 are saying as this financial year draws to a shaky close.

While the U.K.'s ailing computer manufacturers were getting all the sympathy, the money from the Christmas selling period was trickling back to the software houses from the distributors. As Oric crashed, and Trade and Industry Secretary Norman Tebbit met Olivetti to hear some soothing reassurances about Acorn's future, British software makers were totting up the cost of staying in the business for another year.

With their pocket calculators going through batteries like C-5s, let's do some sums of our own to see what's scaring them. First, how much is the British software market worth? Around £85 million from the retail end sounds about right. But of that only about £38 million will go to the software houses.

Next, how many companies are fighting for a slice of that £38 million cake? Back in the heady days of 1983 there were probably more than 300 software producers. Many were one man and a dog and program outfits. Now as the market punishes the unprofessional fewer than 150 remain. And only 100 of those have had any measure of success.

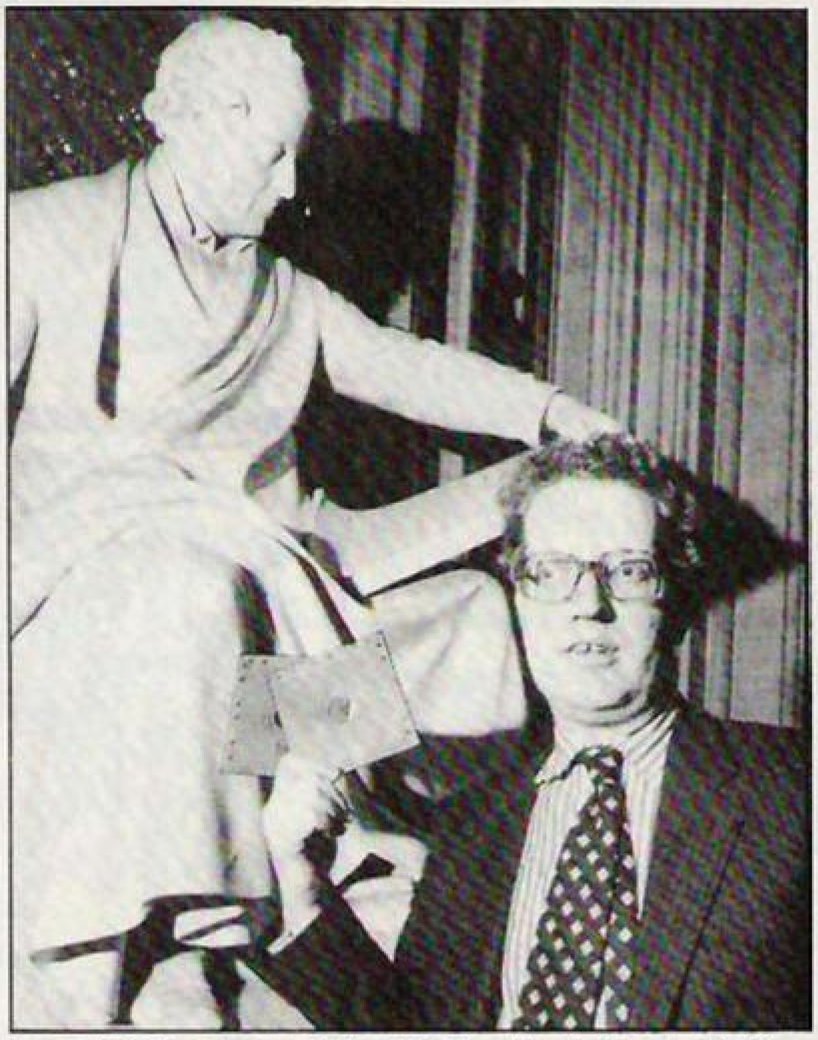
Well, £380,000 revenue per software house doesn't sound bad, but the market research company Ram-C reckons that the top 30 producers accounted for 87 per cent of the takings in the crucial Christmas period — with the top three of Ocean, Ultimate and U.S. Gold scooping up more than 30 per cent.

That leaves 70 firms scrapping it out for 13 per cent of £38 million — just over £70,000 each assuming they each get an equal share, which they won't. Now deduct running costs like salaries, royalties, tape duplication, marketing and advertising, and the word "shake-out" will be flashing up on LCDs all round.

Who is to blame and can anything be done? Perhaps the software houses themselves should take some of the criticism for not making the right quality of innovation last year when home computing was reaching a crucial point in its growth.

Price cutting may help in the short term but the real hopes are pinned on the new powerful computers due this year from Atari and Commodore to give the software market a badly needed fillip.

Editor: TOBY WOLFE; **Assistant Editor:** MEIRION JONES; **Production Editor:** IAN VALLELY; **Software Editor:** SIMON BEESLEY; **Commercial Software Editor:** PAUL BOND; **Editorial Assistant:** LEE PADDON; **Editorial Secretary:** LYNN DAWSON. Editorial: 01-661 3144. **Advertisement Manager:** NICK RATNIEKS 01-661 3127; **Senior Sales Executive:** JULIAN BIDLAKE 01-661 8458; **Advertisement Executives:** NIGEL BORRELL 01-661 3660; KAY FILBIN 01-661 8484; **Northern Office:** GEOFF PARKER 061 872 8861; **Advertisement Secretary:** MAXINE GILL; **Classified:** SUSAN PLATTS 01-661 3036; **Publishing Director:** CHRIS HIPWELL; **Group Advertisement Manager:** SHOBHAN GAJJAR. Your Computer, Room L221, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS. Printed in Great Britain for the proprietors Business Press International Ltd, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS. Tel: 01-661 3500, Telex/grams: 892084 BIPRESG. ISSN 0263-0085. Printed by Riverside Press Ltd, Whitstable, Kent, and typeset by Instep Ltd, London EC1. Subscriptions: U.K. £12.50 for 12 issues. ABC 154,334 January-June, 1984.



Powell fights piracy while old friend looks for dandruff.

PIRACY LAW NEAR

WILLIAM POWELL'S MP's anti-piracy Bill should be law by July. The Amendment to the Copyright Act will remove any legal obstacles for software houses which want to prosecute counterfeiters and home tapers, and will allow courts to impose fines of up to £2,000 or two months imprisonment for selling, exhibiting or even just possess pirated software.

Counterfeiters themselves could face two years in gaol or unlimited fines. The Federation against Software Theft which has promoted the bill is now co-operating with the Federatin against Copyright Theft which co-ordinated raids on video counterfeiters after that part of the copyright law had been clarified to cut video piracy by two thirds.

Ghost busters

REMEMBER Captain Spectre — the eccentric Cambridge spook hunter who hoped to bust ghosts with his micro? What he needed was a Banana.

Banana is what Castle Associates of Scarborough, 0723-584250, are calling their new interface for the Commodore 64 and BBC. It costs around £230 and can handle eight inputs such as temperature, touch, movement, sound and light sensors and then drive eight outputs controlling cameras, lights, recorders, even nets and trap-doors if you must.

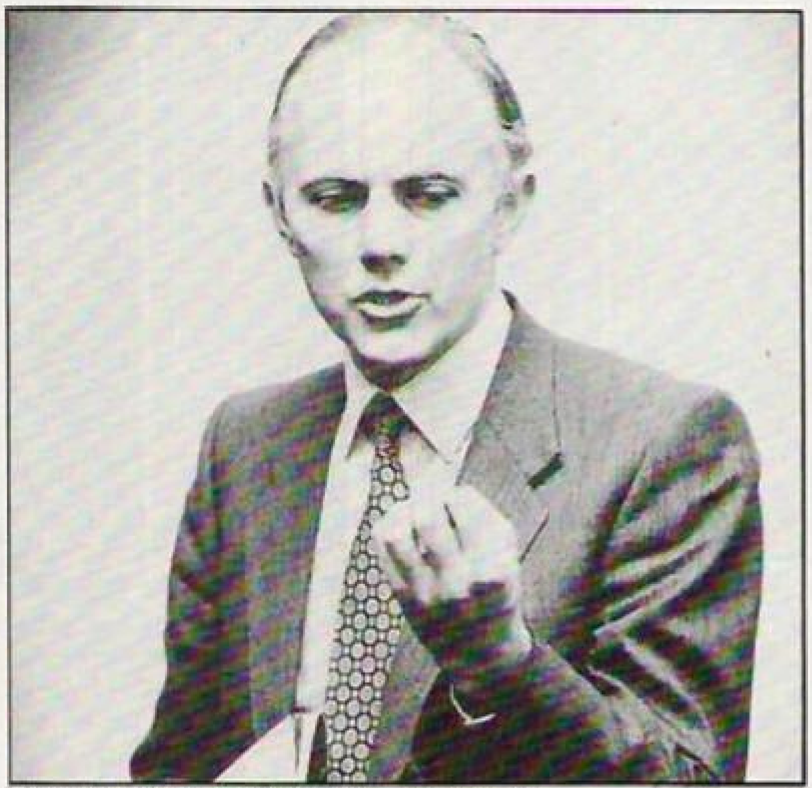
Why Banana? As it is designed for students Castle makes it "so simple a monkey could use it".

ACORN SURVIVES
Olivetti money to the rescue

"MORE LIKE the Turin shroud than a saviour" was how one Cambridge employee described Olivetti's purchase of 49 percent of the ailing BBC Micro makers.

With £47 million of debts and the share price crashing from 193p to 28p before trading was suspended in February Acorn was in desperate need of a rescue operation. Poor Christmas sales and £10 million losses on the failed attempt to break into the American market had created the crisis.

Now Italian typewriter and micro maker Olivetti has bought nearly half of Acorn for £10 million with an option to make that a majority shareholding, but the BBC will keep to their deal with Acorn for the moment. Alex Reid, appointed Chairman in the emergency over Chris Curry and Herman Hauser's heads admits that "we will not have an easy ride over the next 12 months". But he hopes that restructuring the company into four divisions, Education, Consumer, Scientific and Industrial and Business and sacking 120 staff will allow Acorn to survive.



Alex Reid: favourite food — spaghetti.

The Electron will still be sold — perhaps in a pack with a disc drive — but there are big stockpiles so it may never go back into production. The ABC range will now just be sold to the specialist research and development market and there is still no word as to when the BBC will be replaced and what with.

SINCLAIR'S SECRET WEAPON — THE QL V2



QL V2 looks the same as V1.

QL OWNERS who did not take up the £35 option to join the Qlub are angry that Sinclair expects them to pay for the new get-it-right Version 2 of the free Psion software.

Sinclair's issuing of Version 2 which loads twice as fast, occupies less memory in the machine and operates 25 percent faster than Version 1 is a tacit admission that the Quill, Abacus, Archive and Easel programs were flawed in their original QL implementation. In effect Sir Clive is relaunching the QL with some software and peripherals finally available and others coming soon.

After the disastrous 50,000 sales in the first year of production Sinclair has publicly announced a target of 200,000 QL sales in the U.K. alone in 1985.

The QL now has a communications package Q Com which includes a modem, auto dial/answer unit and interface.

Computamate has produced a Q-Disc floppy-disc controller and Computer One has completed QL Pascal, Forth, Assembler and Monitor. Other companies have produced BCPL, LISP, graphics, accounts, and filing packages but games are still limited to chess, backgammon and bridge.

Soft Aid £200,000 Ethiopia target

SOFTAID — the £5 "Greatest Hits" games tape for Ethiopia is now on sale. Rod Cousens of Quicksilva, who organised the appeal, hopes that it will raise £200,000 for the Bob Geldof Band Aid Ethiopian Appeal Fund.

The tape, which includes Kokotoni Wilf, Ant Attack, Horace goes Skiing and 3D Tank Duel amongst others, and an audio track of Band Aid's *Do they know it's Christmas* is available in Spectrum and Commodore 64 versions.

1985 Grand National on Spectrum

JUST BEFORE the real race begins, Elite has launched Grand National for the Spectrum, which lets you place your bets without risking any money and then watch overhead and side views as your horse completes the 30 fence Aintree course. You steer the reins, crack the whips to increase speed and time your jumps, but foul play can lead to a steward's enquiry. The program costs £7 but you might be better off putting that on Lucky Vane.

Coast to coast by C5

SIR CLIVE is about to risk his C5 on a John O'Groats to Land's End run. 28 days?

Meanwhile, Sinclair has been spelling out his plans for successors to the C5 electric triangle. The C10 will be a side-by-side two-seater with 30mph top speed and 40 mile range and should be on the roads in 28 months, while the much more ambitious four-seater teardrop-shaped C15 capable of 80mph flat out for hundreds of miles, will not appear until the end of the decade.

LIGHTMAGIC GRAPHICS DESIGNER



£14·95

Lightmagic is the ultimate graphic design package giving you the ability to create stunning graphics and complete illustrations at your fingertips.

Now you can produce and edit complete screens and pictures. Use Lightmagic as a fascinating program in itself or as an exciting new dimension in your games and programming.

Just look at these features!

*Fully integrated screen editor and UDG designer.
*Drawing with pens.
*Painting with brushes – 10 different effects in 20 sizes.
*Illustrating with airbrush.
*Cut and paste facility.
*Lettering in upper and lower case characters, also double height facility.
*User Defined Graphics – Choose from 147 pre-set graphics or define your own.
*UDG Designer for creating and editing up to 7 sets of User Defined Graphics.
*Facilities to mirror, invert, copy, move, enlarge, fill, rotate, etc.

A comprehensive manual with worked examples, gives you a step by step guide to Lightmagic through every stage of screen composition.

Full facilities are provided for tape storage of finished pictures and UDG's, together with a temporary memory storage facility.

So enter Lightmagic and start designing today!

For 48K Spectrum and Spectrum +

The Complete Machine Code Tutor



£14·95

Specially written for BASIC programmers finding the prospect of machine code rather daunting!

Now there's a straightforward way to learn all about machine code and its many applications.

Here are just a few of Machine Code Tutor's many features:

*A complete machine code tutorial on two cassettes with up to 100K of data depending on machine type.
*Between 27 and 35 progressive lessons depending on machine type.
*Full explanatory simulator – displays relevant registers and flags and their effect on the processor.
*Runs a series of exercises for continuous assessment of your progress which can be modified or substituted.
*No matter how much you experiment the safeguard system prevents crashing of the program.
*Automatic error checking.
*All machine code instructions are dealt with throughout the program and in the accompanying manual.
*Comprehensive reference manual.

Machine Code Tutor really is the most complete instructor around. Learn from your computer rather than a book, and do it at your own pace.

Using the tutor you should be confident of writing complex machine code programs by the time you finish your last lesson.

For 48K Spectrum and Spectrum + Commodore 64 BBC Model B 32K Atari AND NOW Amstrad CPC 464

New Generation Software

Lightmagic and Machine Code Tutor are available from all good stockists, or direct by filling in the coupon below and returning to:
New Generation Software Ltd., FREEPOST, Bath BA2 4TD. For instant despatch phone your order now on 0225 316924 quoting your Access or Barclaycard number.

Please send me ______ Lightmagic(s) at £14.95 each, and ______ Machine Code Tutor(s) for the ______ Computer at £14.95 each. P&P FREE.

I enclose my cheque/postal order made out to New Generation Software Limited for £ ______ Cheque/Postal Order No. ______

Name ______ Address ______

Card No. ______ Signature ______ New Generation products are sold according to their terms of trade and conditions of sale.

VISA

YC/4

IS ACORN'S ALEX REID really Mr Chessman in disguise? Chessman was the strange food-obsessed villain of a recently repeated Avengers episode. Famous scientists and electronics specialists kept disappearing into his very English hotel only to find that they had been sold to a foreign power. Chessman's only weakness was that he would die if the central heating was turned down below 80°. Imagine my surprise at an Acorn "don't panic" press conference after the Olivetti deal which was held in a room with the temperature in the nineties. Nevertheless, Reid refused to enter the room until his aides had closed all the windows. One of his PR cronies sat in a corner trying to hide a file marked Top Secret. Inside was a list of "Danger Questions" such as, "What would the foreign power do if you didn't succeed" to which he gave the prepared answer "kick me out". The one question the PR had failed to predict emerged, "What if you do succeed?" A strange Chessman-like gleam came into his eyes and Reid answered, "I will retire with a large plate of spaghetti from Olivetti in 2001".

"WHY SHOULD the devil have all the best computer games" asks the Reverend Stoker Wilson — bringing General Booth up-to-date. He and his 900 strong Church Computer Users Group are going forth and multiplying and now publish a Top Ten of christian programs. No. 1 is Jericho Road with 2300 sales. "Two years ago it was just one or two freak vicars" says Stoker. I remind him that he wrote to us in 1983. No checks are made on the religious bona fides of members. "So they could be devil-worshippers", I ask innocently, perhaps thinking of Bram Stoker. "They could be", Stoker replies, "but you generally find they're not".

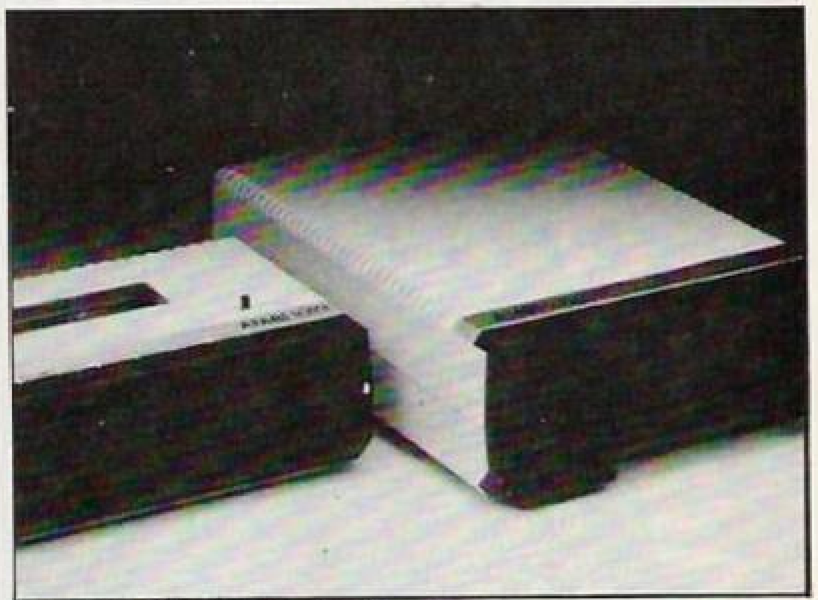
SOME CHARTS measure "not what the shops are selling but what they would like to get rid of" mutters Riva Gould, adding darkly something about people getting places in the Top 20 in return for "favours". Not the church chart I hope. Riva used to promote Camputers. "I used to see the Lynx selling very well — when I knew it wasn't."

Elsie Dee

PRICE CRASH
Free disc drive offer

ATARI'S new 64K computer plus disc drive offer for £250 sets the pace in the price war. But Commodore's halving of the Plus 4's £300 tag will also make competition rough on the £130 Spectrum Pluses and Electrons.

The Atari offer is £130 800XL with a £200 1050 disc drive and two discs containing a home filing manager and demonstrations all for £250. The Plus 4 at £150 is a 64K computer with simple spreadsheet, word processor, database and filer all built in on rom, but there are fears that Commodore may soon abandon the product without software support. Commodore denies this but with a whole range of products — from the CBM128, through the 68000-based AMigo, to the PC and Z-8000 all due to be launched this year the range is beginning to look crowded — especially now that the CBM-64 has been reduced to £160 yet still finds two Commodores, Plus 4 and 16, or both must go. Meanwhile, Orics are being sold for £50 or less and some stores are selling off BBC model A's for £150 or less — cheap if you can find an inexpensive upgrade.



From tape to disc is Atari's way.

HAL RETURNS FROM DEAD



Can Hal survive till 2019?

SPACE AND COMPUTERS are mixing happily at the moment. While Douglas Adams's Hitchhiker's Guide to the Galaxy computer game has finally knocked Flight Simulator off the top of the American charts, over here HAL-9000 the oversized calculator that became a film star in *2001*, is returning to our cinemas in *2010*. Peter Hyams, who has adapted Arthur C Clarke's novel for the screens wants to make the computer a little less threatening than he was in 2001. A joint Soviet American mission is travelling to Jupiter on the spaceship Leonor. En route it has to encounter the Discovery, which is still in the grip of that on-board computer Hal, which has to out-act Sal9000, a red she-computer, Helen Mirren and a bunch of assorted Americans and Russians if it is to survive.

Blame the weather on Mac

BBC's NEW weather presentation uses an Apple Macintosh to display the overlays for the chart. Clouds, sunshine, snow, rain and hail are all icons which can be moved around the screen with a mouse.

In preparation for the change the BBC experimented with joysticks and graphics tablets but found that on the mouse with icons was the best solution.

Chess Game it is not

WHETHER Karpov ends up fighting Kasparov in London's docklands this year or not, MicroClassic will go ahead with the launch of The Chess Game for the Commodore 64.

For the game is not chess but an arcade adventure based on the theme of the game. Up till now MicroClassic has worked as a development house producing programs like Hunchback for Ocean.

SINCLAIR
ACT, COMMODORE
ACORN, MSX, ATARI, ETC

MICRO EXPORT

We export Microcomputers and Peripherals to all Countries
Any make. Any Quantities
BIG DISCOUNTS

send for comprehensive price list to:
Dept W
MICRO-X LTD
765-767 HARROW ROAD, LONDON NW10 5NY
TELEPHONE 01-968 6622 TELEX 915866 MICROX G

RSD Connections Ltd

CONNECTORS IDC

	Card Edge	(Transition) PCB	2 Row Socket
10 way	£1.20	£0.85	£0.80
14 way	—	—	£0.90
16 way	£1.60	£1.20	£1.00
20 way	£1.90	£1.35	£1.20
26 way	£2.40	£1.60	£1.45
34 way	£3.10	£1.95	£1.60
40 way	£3.40	£2.00	£1.85
50 way	£3.85	£2.25	£2.00
60 way	£4.80	£2.60	£2.25

IDC SHROUDED HEADERS WITH EJECTING LOCKING ARMS

	Straight	Right Angle
10 way	.85	.90
14 way	£1.00	£1.20
16 way	£1.25	£1.45
20 way	£1.40	£1.60
26 way	£1.70	£1.95
34 way	£2.00	£2.30
40 way	£2.15	£2.45
50 way	£2.30	£2.65
60 way	£2.45	£2.85

IDC D CONNECTORS

	Male	Female
9 way	£2.70	£3.20
15 way	£3.20	£3.70
25 way	£3.80	£4.45
37 way	£5.90	£6.80

IDC AMPHENOL

(Centronics)

36 way plug	£4.95
36 way socket	£5.95

PRINTER LEADS

ALL 1 METRE LENGTH

BBC	£9.95
BBC serial	£8.95
AMSTRAD	£16.75
MEMOTECH	£12.75
EINSTEIN	£12.75
IBM PCC	£15.95
SPECTRUM interface 1 to RS 232	£9.75
SPECTRUM interface 1 to Centronics	£10.75
CENTRONICS to Centronics	£15.20
DRAGON	£9.95
QL	£10.75

COMPUTER LEADS

Dual disk drive 1m	£10.25
Single disk drive 1m	£7.25
7 pin DIN plug to 2x3.5mm 1x2.5m Jack plugs	£2.26
5 Pin DIN plug to 2x35mm 1x2.5mm Jack plugs	£2.20
6 Pin DIN Plug to 6 Pin DIN Plug	£1.95
2x3.5mm Jack plugs to 2x3.5mm Jack plugs to	£1.25
7 Pin DIN plug to 5 pin DIN plug	£1.95
7 Pin DIN Plug to 3 pin DIN plug and 2.5mm Jack plug	£2.25
7 Pin DIN plug to 7 Pin DIN plug	£2.25
5 Pin Domino Plug to 5 pin Domino plug	£1.95
Phono Plug to Coaxial TV Aerial plug	95p

D CONNECTORS

Solder Bucket	Male	Female
9 way	.75	£1.00
15 way	.95	£1.50
25 way	£1.50	£1.90
37 way	£2.40	£3.25
HOODS .95		

RIBBON CABLE (Price per ft)

	Gray	Rainbow
9 way	.13	.24
10 way	.15	.28
14 way	.18	.32
15 way	.20	.36
16 way	.22	.40
20 way	.30	.50
24 way	.36	.60
25 way	.40	.65
26 way	.45	.70
34 way	.60	.80
37 way	.65	.85
50 way	.70	.90
50 way	.95	£1.25
60 way	£1.10	£1.40
64 way	£1.15	£1.50

DIP HEADERS

Pins	
14	90p
16	£1.00
24	£1.40
28	£3.95
40	£2.15

EDGE CONNECTORS

2x23 way (ZX81)	£1.85
2x28 way (Spectrum)	£2.10
2x6 way Commodore	£1.20
2x12 way	£1.60

DOUBLE SIDED PLUG BOARDS

ZX81 23 way	£1.25
Spectrum 28 way	£1.50

TELEPHONE CONNS

Surface master jack socket	£3.75
Surface extn socket	£2.50
Dual outlet adaptor	£4.25
Line jack cord 3mtr	£1.85
BT jack plug	48p
4 core cable per metre	15p

All surface units shuttered BT approved.

MONITOR LEADS

QL to Ferguson	£2.50
BBC to Ferguson	£2.50
BBC 6-pin to scart plug	£4.95
BNC plug to phono plug	£2.95
QL to scart plug	£4.95
QL to mono monitor	£1.25



BBC JOYSTICK DELUXE JOYSTICK for BBC MICRO. Fits directly into BBC Micro analogue port

£12.95

SPECTRUM EXTENSION LEADS

6 inch extension calbe M to F	£10.50
6 inch F to 2M's	£14.75
12 inch extension cable M to F	£10.75

Other convigurations made to order
Write or phone for quote

CUSTOM ASSEMBLIES

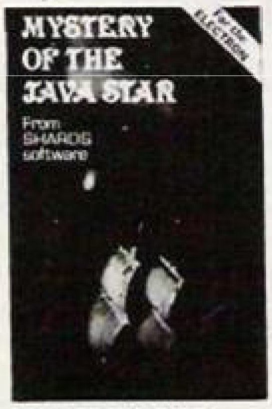
ANY COMBINATION OF LEADS MADE TO ORDER

TRADE ENQUIRIES WELCOME

Please add 15% VAT *Free p&p C.W.O. or use*
RSD Connections Ltd., Dept YC2, PO Box 1, Ware, Herts.
Formerly COMPUTER LINK UK LTD. **Telephone: 0920 5285**

SHARDS

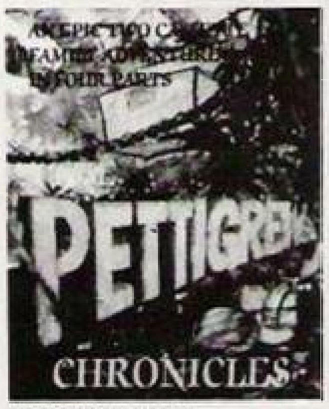
THE FAMILY FAVOURITES



ELECTRON, BBC
DRAGON/TANDY COCO
CBM 64/AMSTRAD (March)
£7.95

FOUR PART EDUCATIONAL ADVENTURE

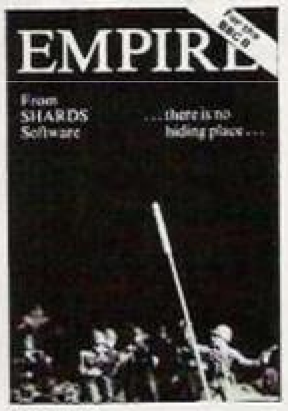
"Shards are now setting a standard for this type of adventure."
Micro Adventurer.
"An excellent educational program with something of interest to kids of all ages – including big ones".
Electron User.



SPECTRUM 48K,
CBM 64
£9.95

EPIC FOUR PART FAMILY ADVENTURE.

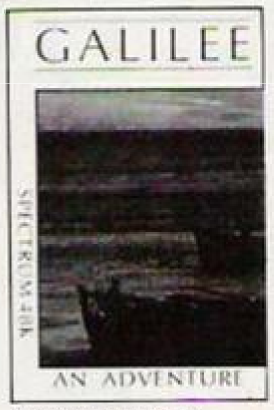
"I liked this mixture of puzzles, games and text adventure and felt it represented good value for money."
HCW.
"An entertaining release ... a nice package to play together after Christmas lunch". PCW.



CBM 64,
ELECTRON, BBC
DRAGON/TANDY COCO
£6.95

COLOURFUL GAME OF WORLDWIDE STRATEGY.

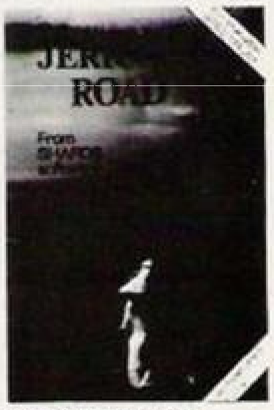
"I thoroughly enjoyed this game... the graphics are excellent".
(Electron) HCW
"Not a bad attempt at simple war-gaming ... a good game".
(CBM 64) HCW.



SPECTRUM 48K
£5.75

EXTENSIVE ILLUSTRATED TEXT ADVENTURE.

"An intriguing adventure, original in many ways... should appeal to most adventurers."
PCN.



SPECTRUM 48K
BBC/ELECTRON (March)
£5.75

FULL FEATURED EDUCATIONAL ADVENTURE.

"A gentle, original and enjoyable adventure".
Sinclair User.
"The program is enjoyable and interesting because of its historical accuracy".
PCN.

AVAILABLE NOW FROM SELECTED BRANCHES OF BOOTS AND ALL GOOD STOCKISTS
or by mail order, using the coupon.

To: SHARDS SOFTWARE, 189 ETON ROAD, ILFORD, ESSEX, IG1 2UQ (phone 01-514 4871)

Please send me a copy of (please tick) Pettigrew Chronicles ☐
Empire ☐ Galilee ☐ Jericho Road ☐ Mystery of the Java Star ☐
My computer is ....
Name ....
Address ....
....
Please debit my ACCESS/VISA Account, A/C No. ....
(or phone 01-514 4871 to save time) I enclose a cheque/PO remittance for £ .... made payable to SHARDS SOFTWARE.
Signed ....

POP★ TOPS

SOFTWARE TOP 20

↑	1	Matchday	Ocean	
↑	2	Booty	Firebird	Sp
↓	3	Ghostbusters	Activision	Sp
↓	4	Daley Thompson's Decathlon	Ocean	Sp
*	5	Brian Bloodaxe	Edge	Sp
↑	6	Airwulf	Elite	Sp
*	7	Starstrike	Real Time	Sp
—	8	Knight Lore	Ultimate	Sp
*	9	Technician Ted	Hewson	Sp
*	10	Automan	Bug Byte	C64
↑	11	Wild Bunch	Firebird	Sp
*	12	Frak	Statesoft	C64
↑	13	Blockbusters	Macsen	Sp
*	14	Go Go Ghost	Firebird	C64
↑	15	Overdrive	Superior	El
*	16	Blue Max	US Gold	Sp
↓	17	Gift From The Gods	Ocean	Sp
—	18	Skool Daze	Microsphere	Sp
↓	19	Combat Lynx	Martech	Sp
↓	20	Elite	Acornsoft	El

Sp = Spectrum/Bc = BBC/16 = C16
El = Electron/Vc = Vic-20
Dr = Dragon/64 = CBM64
Am = Amstrad/Or = Oric/At = Atari
* = New release — Same position
Source: WH SMITH

SOFTWARE TOP 3 BY MACHINE

↑	1	Punchy	Mr Micro	Vc
↑	2	Perils of Willy	Software Projects	Vc
↓	3	Psycho	M. Tronics	Vc
↑	1	Booty	Firebird	64
↑	2	Ghostbusters	Activision	64
↑	3	Daley Thompson's Decathlon	Ocean	64
↑	1	Booty	Firebird	Sp
↑	2	Airwulf	Elite	Sp
↑	3	Ghostbusters	Activision	Sp
*	1	Chuckie Egg	A&F	Dr
↑	2	Dragon Chess	Oasis	Dr
↑	3	Frogger	Microdeal	Dr
↑	1	Millionaire	Incentive	Bc
↑	2	Magic Sword	Database Pub	Bc
↑	3	Sports Quiz	Kosmos	Bc
*	1	Encounter	Hi-Tech	At
↑	2	Attack of The Mutant Camels	Llamasoft	At
↓	3	Zaxxon	Centresoft	At
↑	1	Flight 015	Craig Comm.	16
↑	2	Games Tape 1	Melbourne Hse	16
↑	3	Williamsburg Adv.	Microdeal	16
↑	1	Elite	Acornsoft	El
↑	2	Overdrive	Superior	El
↑	3	Felix Meets Evil Weevils	Micropower	El
*	1	Football Manager	Addictive	Am
↑	2	Manic Miner	S. Projects	Am
↑	3	Snooker (S. Davis)	CDS	Am

Source = Websters Software

YOUR Letters

TELSOFT IN NZ

TELSOFT works very well from here in New Zealand. We have downloaded four programs sucessfully now. The phone bill is not excessive either as it doesn't take too long to download the programs.

We are using a VTX 5000 modem. One suggestion we do have, though, is that we would like to see more complete instructions on how to save off the more complicated programs like Windos. This is the only program that we have not been able to run.

D. T. Ready,
Manukau Computers (NZ) Ltd.,
Auckland,
New Zealand.

ZX-81 EXCHANGE

YOUR CORRESPONDENT S.P. Kaliszczak — February 1985 — is by no means alone with his problem, ie finding software for the ZX-81. Through the postal user group, ZX Exchange, of which I am organiser, this is perhaps the most frequently voiced complaint of recent months.

The solution, as I continually advise people, is to write your own, and then to exchange it with others. This is not as difficult as it may seem provided that the right help is forthcoming and that is exactly the purpose for which ZX Exchange exists. By this kind of mutual co-operation, the ZX-81 user not only gets programs tailor-made to his own requirements, he is also forced to learn the fundamentals of computing, but in an agreeable and friendly manner. If and when he decides to acquire a more versatile and expensive machine he is well equipped first to get the right machine for his purposes and, second, to make the best of it when he gets it. Any of your readers who are interested in ZX Exchange are invited to write to me with a s.a.e. at the address below, or to telephone me any evening.

Since I am writing, however, I must take the opportunity to congratulate *Your Computer* on having continued to publish ZX-81 programs of considerable quality, despite the obvious temptations to forget about this "downmarket" area. Certainly, the number and quality of such programs has proved sufficient to keep me as a regular subscriber long after I have given up other magazines; long may that situation continue!

Nick Godwin,
4 Hurkur Crescent,
Eyemouth,
Berwickshire,
Scotland,
TD14 5AP.
Telephone: Eyemouth (0390) 50965.

"BASIC IS BEST"

IT SEEMS to be the fashion these days to blame Basic and to praise any other languages, like A. Hegedus, D. Sutton — *Your Computer*, October 1984 — and A. MacPhee — *Your Computer*, January 1985 — do in their letters. That kind of talk can only be understood as an attempt to give the impression of being a computer expert but, in fact, it shows ignorance of programming and Basic language.

Standard Basic is a structured, modular and extensible multi-tasking language. It is powerful, but still easy to learn and use. Of course, the Basics in most cheap home-micros are very

(continued on page 35)

SINCLAIR TRANSPORT KEEPS ON KEEPIN' ON



RECENTLY, WHILE researching into my own family history, I came across the following which alludes to the inventiveness of a Sinclair, approximately 100 years ago. He is refered to in 1886 as being, "the present Earl of Caithness" who must be the 14th Earl, James Sinclair.

Some similarities between the "steam car" of Sir James and the "electric" one of Sir Clive deserve examination and our admiration.

D. C. Hardy,
Torquay,
Devon.

"The front view of the vehicle is that of a phaeton placed on three wheels, and made a little wider than ordinary, so as to have room for three or even four abreast. His Lordship sits on the right hand side and drives, resting his left hand on a handle at the end of a bent iron bar, fixed below the front spring, to the fork in which the front wheel runs, and guiding the direction of the carriage.

"Placed horizontally before him is a small fly-wheel, fixed on an iron rcd, that, passing downward, works at the lower end by a screw, through one end of a lever, attached at the other end to a strong iron bar that passes across the carriage, and has fitted on it a drag for each of the hind wheels. By giving the fly-wheel in front a slight turn with his right hand, his Lordship can apply a drag of sufficient power to lock the hind wheels and stop the carriage on the steepest declivities of common roads.

"Inside the carriage, a line backward from his right hand, is placed a handle, by which the steam is let on, regulated, and shut off at pleasure.

"The power of the engine, and the perfect control His Lordship has over it, enabled him on several occasions to make long journeys over rough and mountainous roads at the rate of eight miles an hour; there can therefore be no doubt that carriages propelled by steam can be used for the purpose of traffic on common roads.

"A journey of 140 miles made in two days, at a cost of less than one penny per mile for fuel, proves this; and the fact that no accident to man or beast was caused by the steam carriage during the whole journey, answers the objections as to frightening horses.

"His Lordship continues to use the carriage, and is most kind and courteous in explaining its constructing and working."

IN ★ TOUCH

How to write for *Your Computer*

We called this magazine *Your Computer* precisely because we welcome your views, tips and hints and even your criticisms of machines and software in general. Here's how you go about getting your name into print. Your article should be typed, double-spaced, on A4 paper. A name and address on each sheet would help. Don't forget to tell us which machine it runs on. With programs please include a cassette or disc and some indication of how long it is. Please put what machine it's for on the envelope. Don't forget full instructions to us how to load and list your program and how to enter it for the readers.

The article must be submitted exclusively to *Your Computer*. We pay £35 per published page — that's as it appears in the magazine and includes illustrations.

Telsoft

Telsoft is *Your Computer's* software downloading service. Any program for the Spectrum or the BBC — and soon the Commodore — which has a telephone symbol next to it is available on the service. Both 1200 and 300 baud speeds are catered for. For more details call Colchester (0206) 8068. No more lonely nights typing in endless pages of hex digits.

Message service

You can get messages to us in two ways. Either use the Prestel Telex Link to 892084 BISPRES G or you can use our very own modem, day or night on 01-661 8978. The modem is V21, 300 baud, even parity, 10 bits per character. You simply transmit in upper case "YRC" — our address code, and wait for the acceptance code "+ + + STF GO". Then off you go. Don't forget to tell us who it's for. Sign off with "NNNN" — again in upper case.

YOUR Letters

(continued from page 33)

cut-down versions, but even those are in many aspects better than the standard versions of Pascal or Forth. And comparing £10 Basic to £300 Pascal is not very fair, is it?

In my work, I have used several languages, including Pascal, Fortran, PLZ/SYS, C, Assemblers and, of course, Basic. I have often made comparisons by writing the same program or procedure in different languages. In almost all cases programming in Basic has been the easiest, the shortest and the most readable. I suspect that those who criticise Basic have never done such comparisons.

Perhaps the most serious claim that has been made against Basic is that it leads to bad programming practice. However, the reason is not Basic, but the fact that home computer manuals do not teach programming, they only list the commands. How would you expect anyone could learn programming without any aid? If you try with Pascal or Forth, the result is at least as bad. The only difference is that those languages are more difficult to learn.

Now that the ANSI Basic standard is finally coming, I believe that Basic will be more widely used and more appreciated in the future.

Pauli Lindgren,
Software engineer,
Helsinki, Finland.

"GET ORGANISED, PSION!"



AFTER WINNING a Psion Organizer in a competition in *Your Computer* last September and receiving it in January, I set about using it. A fun little thing I thought but what about a practical application. Well, in the competition I wrote down I would use it to keep records of my cows. So that's what I set about doing.

The Organizer comes with a Utility Pack and a 5K pack which allows you about 12,000 characters which would be inadequate for my herd. But you can buy packs of 14K. I then worked out a form to use and decided two 14K packs would be sufficient for five years supply of data.

The Organizer has a great ability to cross refer and return data which is exactly what I need. The format I devised would have to be able to find a given number, such as a cow's number, eg No. 53 and also tell one when she calved, if the vet has seen to her and when she was served by the A.I. etc. These records must be able to be cross filed with 100 other cows. So I could call for all the cows that calved in August to be printed out etc. This format would need to be updated at least five times a year. With the Organizer when you update data you delete or add to a file, then resave it.

The format I went for ran as 53 — CAUG 14/4 — VSEPT 10/4 etc. This allowed me to save bytes and cross file. Type in 53 and I get the whole record for that cow, or Type *C* (stands for calving) *Aug* (date) and I get every cow which calved in August. The 14/4 is the *14*th day *1984* and so on.

As you can see the range of data is vast. So off to Bristol I went. W.H. Smiths — "no". Boots — "yes". "Two 14K packs please", "£39.90" came the reply. "Hell, no thanks" and home I went.

Here I have a computer to use with a practical application but cannot afford the software. The Organizer is a very powerful compact unit with mathematical and scientific use. The data handling is excellent with a fast multi cross filing system but 20 quid for a 14K pack?

Rod Shaw,
Nailsea,
Bristol.

VISIBLE ERROR

THANK YOU for publishing the Invisible Toolkit. A typing error appears at the bottom of page 111 — *Your Computer*, March. The address 16544 should read 16514. My thanks to George Payne of Watford for his letter. Also, here are a few modifications to improve the handling of system variables automatically:

```
20 LET L = 768
405 POKE 16417 , 0
415 IF PEEK 16417 = 1 THEN POKE
    16384 , 28
```

Change all occurrences of the line number 10222 to 10220.

Before running program 3, type in the following pokes:

```
17327,6
17328,253
17329,203
17330,33
17331,198
17332,24
17333,144
17334,201
```

All cassette tapes are the modified program.

Stuart Clark,
Glenrothes,
Fife.

BULLETIN

I WOULD APPRECIATE it if you would include my Bulletin Board, the first one in Scotland, in your magazine. My Bulletin Board is called SABBS (Scottish Atari Bulletin Board Service) and is on 0698 884804. It runs 24 hours a day at 300 Baud on an Atari 800 XL.

Nick Rosser,
Larkhall,
Lanarkshire.

SECURIFILE

SEVERAL READERS have written to me about my program Securifile for the Spectrum published in the March issue of *Your Computer*. Their enquiries concern the possibility of operating Securifile on Microdrive. I have produced a Microdrive-compatible version which, at the cost of losing a small amount of data and space, may be operated on Microdrive.

This version is available on cassette from 29 Ashridge Drive, Brichet Wood, St Albans, Hertfordshire for £3.75. Please order as version 2.3. The standard tape version is also available at the same price, order as version 2.1.

Jon Ellis,
St. Albans,
Hertfordshire.

I could do that...

Anyone who plays with the Macdraw program on Apple's Macintosh will be impressed by how flattering it is to their artistic abilities. Even those with no gift whatsoever for drawing can produce pleasing effects. In this respect perhaps the most useful of its facilities in the one which mirrors or reflects a drawing in four or eight different axes. In effect this simulates a kaleidoscope, creating a symmetrical pattern where before was just a random squiggle.

For this month's £15 competition we would like you to furnish your micro with a kaleidoscope facility. It should allow you to draw a figure in the top right hand quadrant of the screen, using the keys to move the screen 'pen' a pixel at a time in any direction. At the same time it should reflect the figure in each of the other three quadrants.

If you own a micro such as the BBC which allows the graphics origin to be defined then your task is fairly simple. First move the origin to the cente of the screen. Now to reflect a pixel into another quadrant you need only to multiply its co-ordinates by −1. Thus pixels in the top left quadrant have negative x co-ordinates while pixels in the bottom right have negative y co-ordinates.

If your micro has a fixed origin — usually at the bottom left-hand corner — you can calculate the reflected positions in the same way and the convert to screen co-ordinates by subtracting the amount needed to move the origin.

No 100 line blockbusters please.

February's I Could Do That asked you to supply a bug-ridden program. Unfortunately it drew only a small response and none of the programs submitted were judged sufficiently interesting to publish. However we are still in the market for unusual bugs and will hold the prize over.

BEGINNERS first bytes

Starting out in home computing? First Bytes is for you. Just write to *Your Computer* with any hardware or software problems, no matter how small or simple.

Sinister art of hacking



HACKERS HAVE been in the news lately, earning themselves rather dubious reputations. Originally a hacker just meant anyone who spent hours of his or her — usually his — life locked in a bedroom with a micro. To begin with, this machine would have probably been some kit micro like the once famous and now forgotten Altair. Or it might have been the still-famous Apple computer, which also began life as a hobbyist's toy.

Hackers developed almost mystical bonds with their machines which they knew backwards and inside out. That was in the early days, when micros were primitive, and required weird and wonderful skills like machine-code programming, and the ability to wield a soldering iron.

But things soon changed. In a rather unsporting move, micro manufacturers began to make their machines easier to use — so easy in fact, that almost anyone could use one. This obviously took away most of the hacker's fun, who delighted in the tricks of the trade that were incomprehensible to outsiders. So hackers began to look a little further afield for challenges. They started to take on the outside world.

At the same time that micros became more user-friendly, the communications side of things began to develop. Using a modem and an ordinary phone, a micro could link up with other micros thousands of miles away. They could also be hooked into rather bigger fry, the giant mainframe computers used in business and government applications.

This proved to be great fun. After all, most of the systems had been designed with the specific idea of keeping unauthorised users out. And it is a well-known fact that it is much more interesting to be somewhere you shouldn't. So some of the hacking community moved on from just writing shoot-'em-ups in machine code to playing what seemed like a real grown-up adventure game.

Hacking became a more familiar idea through the film *War Games*. In it, a young hacker used his micro to ring through all the telephone numbers in a certain area to search out which ones were connected to computers. Having found one, he tried to break into it. The film is about his unfortunate success: the computer he gets into controls America's entire missile system. When, as a result of this, the computer gets too big for its own boots, the world teeters on the brink of nuclear catastrophe.

Needless to say, the all-American boy saves the day in the end. The interesting thing is that shortly after the film was released, a real young hacker succeeded in gaining access to a sensitive U.S. military computer. Since then, there have been a number of reports of people, armed with little more than a micro, a modem and a mischievous sense of fun, getting into several supposedly watertight systems.

In this country, British Telecom's Prestel service has been the victim of hacker's attacks. Abroad, people have got into German financial systems, and computers owned by the French atomic energy authority.

The main way that unauthorised individuals are locked out is by using passwords. These are simply words or sequences of numbers and letters that have to be fed into the computer before you can go on any further. The more passwords a system requires, the more secure it should be. In particular, if there are any operations that are very vulnerable or sensitive, it is customary to demand a special password each time they are used.

Passwords are found on such things as Prestel, Telecom Gold, which is a method of sending private messages by computer over the telephone lines and many small business programs. More advanced programs only allow you a limited number of tries before you are locked out. This is to prevent an obvious hacking technique of programming your computer to run through all the possible password combinations. Not a quick way of getting in, but one that should work — eventually.

You can always try and guess. This is not such a hopeless procedure as it sounds. When it comes to thinking up random passwords, most people are very unimaginative. They tend to pick their own name, or their wife/husband's name, or the name of the dog/budgie, and so on. So a little background information on the person who set the password can often be a good clue. This was in fact the technique used in *War Games*.

Nowadays, though, hackers are a good deal less scrupulous in the means they employ. Using a mole inside the relevant organisation seems to be becoming increasingly common. This is because when passwords are being changed regularly, it should be impossible to crack them consistently. And yet hackers have succeeded in getting into many systems where such precautions are taken. In particular, once a master password has been obtained, not only is it possible to get into a system, it is even quite easy to lock out the legitimate users.

(continued on page 39)

(continued from page 37)

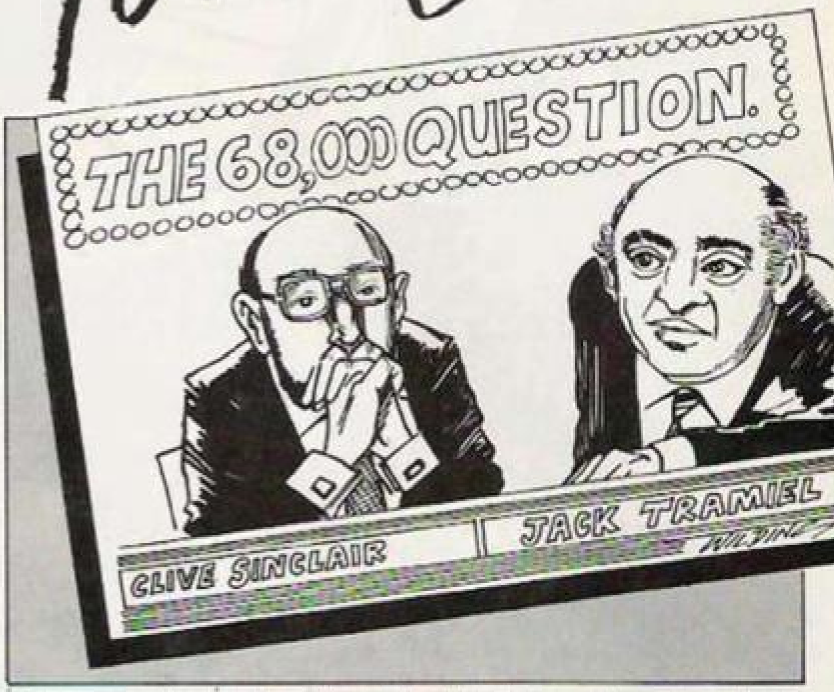
The mole technique succeeds because people tend to be very careless with their passwords. Favourite tricks include leaving them on a scrap of paper next to the machine, or in a nearby drawer. It therefore only requires a little effort to get hold of these precious numbers.

Hacking started as fairly harmless fun by people who relished the extra challenge. But as computers begin to affect more and more of our lives, the dangers of this kind of unauthorised use are growing. For example, in America, a reporter who had written slightly critically of the hacking community found that his credit card number was obtained from his bank's central computer, and was being posted on computer bulletin boards for all to see and use.

It is easy to imagine some of the problems this kind of thing could give rise to. Bank balances could be tampered with, medical details investigated in hospital records, even police records broken in to and possibly changed.

BEGINNERS

first bytes



How and what the processor processes

DECEMBER'S FIRST BYTES described how silicon chips were made. The same basic techniques are used for all the different varieties. This includes the most important chip in a micro, the processor. As its name implies, it is concerned with the actual business of manipulating the strings of digits that are fed into it.

Whether these digits represent blobs moving on screen, Basic programs or word-processing applications, is completely irrelevant. The processor simply takes each command in a program and implements it blindly.

The microprocessor can only understand machine-code instructions, so some translation may be necessary. For example a program written in Basic exists as a series of English-like lines of commands: it is a high-level language — see February's First Bytes for a fuller discussion. If the processor is to obey these commands, they must first be converted to the machine-code instructions that carry out the same operations.

This can be done in two ways. Most Basics are interpreted. This means that when a program is run, each line in turn is converted to machine code by the Basic interpreter. In fact this is precisely what takes up most of the space when you load Basic into your machine. An interpreter has to know all the rules for converting Basic programs into the correct machine code for that particular processor. In addition, it normally checks the syntax of your program as it does this.

Incorrect syntax, which includes things like using words unknown to Basic, leaving off brackets and mixing up the types of variables, will throw up an error message, and stop execution of the program. This is why Basic is such a good language for beginners. It tells you about your mistakes as they occur, so it is far easier to locate and correct them. Earlier, less sophisticated languages simply tried to run your program, failed, and just told you that you had failed — just in case you hadn't noticed.

If this kind of error-checking is a big plus for interpreted Basics, there is also a down side. Since each line is translated and checked and then run one at a time, the whole process is relatively slow. This is particularly true if the program is at all interactive, that is requires constant input from you, as in a computer game. Ideally, you would like to bung in the whole program, and let it run straight away.

This is precisely what an alternative way of running a Basic program does. Instead of interpreting a program, you compile it. This simply means that you feed in all the lines at once to something called a compiler. The program that you enter in this way is called the source code. Once the compilation is complete, you then run the resulting machine code — called the object code.

Such a process breaks up the running of a program into two stages. Provided your program contains no errors, its object code version will run much more quickly, because the processor can just get on with it, instead of constantly pausing to ponder over a line of Basic, translate it using the interpreter, and then run it.

Just as the big advantage of interpreted Basic was its error checking, so the drawback of compilers is that they translate everything before they bother telling you about the errors. This means it is harder to debug a program with a compiler. Ideally you would develop software using an interpreter, and once it is all running smoothly, you would then compile it to a very compact object code file which you would then run instead of the source code itself.

Sometimes professional programmer use this technique when they wish to avoid the trouble of writing in the mind-blowing hieroglyphics of the machine code, but need something faster than ordinary Basic programs.

The end result of interpreted and compiled Basic programs is commands written in machine code. This is true whatever language a program is written in, and for whatever machine. This also explains why machine code represents the ultimate for programmers: they can talk directly to the heart of the micro, the processor.

However this machine code is produced, it is stored in Ram, the temporary memory of your computer. The amount of Ram therefore determines more or less the size and complexity of programs that can be run. More complex programs sometimes use a technique called overlays to allow you to run far larger programs than your Ram can theoretically cope with. For this, though, you need at least a floppy disc system; ideally a hard disc is best.

The technique used is to load as much of a program as possible or is necessary in Ram, and then call in additional chunks as and when they are needed from the disc. This does slow down response times slightly, but can work in the right situation.

The processor works on the machine-code program stored in Ram by obeying one set of commands at a time. Something called the stack ponter keeps a record of where a processor has reached as it works its way through a program in Ram. These commands normally take the form of an instruction, such as telling the processor to add two numbers, followed by data, in this case the two numbers to be added. Alternatively there may be commands that instruct the micro to fetch data from memory locations, and feed them to the processor.

When this is done, data is stored temporarily on tiny memories called registers. These hold the data long enough for the processors to act on them in some way. Similarly, the result of one of these processor operations is stored on a register before it is transferred back to the main Ram memory.

The power and complexity of a processor is partly determined by how many registers it has for storing various pieces of data. The number of basic processor operations, like add, subtract and so on, also varies widely from model to model. But perhaps the most important distinction that can be drawn between various chips is whether they are 8 or 16 bit. This distinction is becoming increasingly important in the home computer world as more powerful machines start to come through.

The numbers 8 and 16 refer to the size of the chunks of data that the processor can move around in any one operation. This includes data that is brought to the registers from memory, or data that is sent from them back to the main Ram. An eight-bit machine moves around eight binary digits at once, in other words a byte at a time. A 16-bit machine can cope with double this, two bytes.

As a very rough and ready rule, this gives 16-bit machines something like twice the speed. Machines that use eight-bit processors include the Spectrum, with a Z-80 chip, and the BBC, which uses the 6502. These two chips together account for practically all of the home micros around. Most business machines like the IBM PC use 16-bit chips, of which the most popular is the 8088.

Things are slightly complicated by the fact that a processor may move blocks of data around in eight-bit pieces, but process them in bigger chunks. Perhaps the most famous of these is the chip used by the Sinclair QL, the 68008. Although internally this super-chip carries out calculations using chunks of data 32 bits wide it can only obtain and send out data eight bits at a time.

Effectively there is a bottleneck of data coming in and out of the main calculating device. The new machines promised by Atari will use a version of the same chip that has a 16-bit data bus. Perhaps they will offer a quantum leap beyond the QL.



If you have anything of interest to tell us about your computer club why not contact us on 01-661 3144.

Local news

Eyemouth

Nick Godwin may be familiar to you as the eminence grise behind ZX Exchange, the postal computer club with a human face. Well, he got an Amstrad for Christmas, and any keen programmers interested in joining a postal Amstrad User Group should contact Mr Godwin at 4 Hurkur Crescent, Eyemouth, Berwickshire, TD14 5AP, or telephone 0390 50965.

Alcester

The Microsoftware Club has worldwide membership as they communicate with each other through their cassette-based magazine. This comes out bi-monthly and is said to be an invaluable aid to anyone who wishes to learn Spectrum programming. They also have their own range of software. Write to Trev Glover, 73 Alcester Road, Moseley, Birmingham 13.

Watford

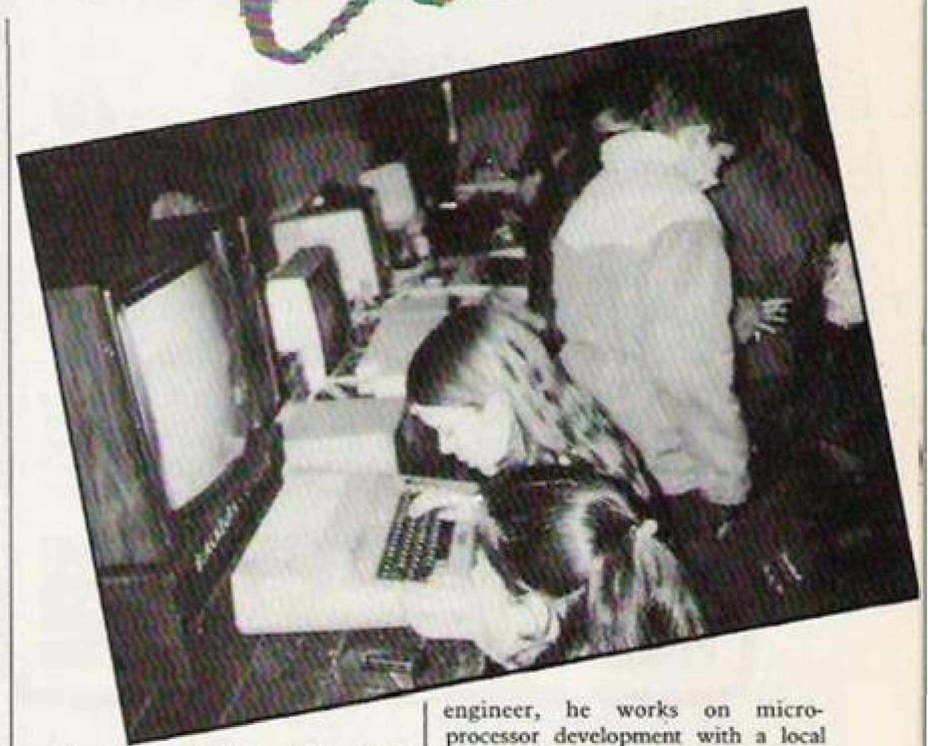
Infomania is a rapidly growing club which has gained over 200 members since its formation in September. Lots of hardware and software is available to members and the club has fragmented into several smaller interest groups specialising in graphics, Logo, the 6502 and Z-80 processors. There is also a Multi User Dungeon (MUD) development group. The MUD runs on an Apricot and can be accessed by Econet on BBCs although it is hoped to go "down the wire" on to modems in the near future. Roger A Burg on (0923) 55122 has details of all the club's activities.

COMPUTER club

Tony Carter discovers that Blackburn's Computer Club, like its football club, is on the way up.

BLACKBURN COMPUTER Club is easy to find. Just across the road from the Darwen end of Ewood Park, home of Blackburn Rovers, is the Fernhurst Hotel; the music room of the large redbrick Fernhurst is the home of Blackburn Computer Club.

Like the Rovers, Blackburn Computer Club is on its way up. On the night that *Your Computer* visited the club most of their 30-plus membership had turned up. The hardware in use included an Atari 800 and an Atari 800XL, both with disc drives, a BBC with disc drive and second processor, a Lynx, an Amstrad CPC-464 and several Spectrums. All this and Thwaites, the native brew of Blackburn.



The aims of the club are to help the members to get the best out of their kit and to give members without a computer an opportunity to try out a variety of machines and software, explained president and founder Bob Hillyard. Certainly everyone seemed to be getting plenty of hands-on experience.

Membership is open to all without age limit. It ranges from children who are at primary school to a 70-year old gentleman who had assembled his own machine. Members include a cheese manufacturer and a market gardener who both use Spectrums to help run their businesses and several families who come to the meetings together.

On this particular night Ernie Troughton and Jimmy Wat were comparing the Tatung Einstein and the BBC B on a value for money basis. Ernie, who is a retired double-glazing manufacturer, owns an Einstein and reckons that apart from the odd review, the Einstein has been ignored in computer magazine articles. A sad fate for a UK-built machine.

Jimmy Wat's job involves him in the world of hi-tec. As a development engineer, he works on microprocessor development with a local firm making machinery for carpet manufacturers.





Deborah, who is 10, likes the club because she gets the chance to try out a lot of different action games. Her sister Heather, who is 12 and her friend Samantha, 13, prefer the children's adventure game club — a regular club feature.

Mark, another 10-year old, was very busy with Pyjamarama on a Spectrum. The club plans forthcoming visits to computer clubs in Chorley and Preston and plans to visit the PCW show at Olympia.

Treasurer George Croft, a Blackburn policeman, made an appeal for educational BBC B software for the Dame Evelyn special school for the mentally handicapped. He also suggested that the club should challenge themselves to collect sufficient funds to present the Dame Evelyn school with a Spectrum.

The club's PR man, John Schofield, handed out part two of the machine-code course he had prepared for interested members and then it was back to the machines again.

The club meets on alternate Monday evenings. Annual membership fees are £3 for under 16s, £5 for adults and £10 for family membership, or members can pay 50p per evening. Members' machines are insured in transit to and from club meetings through the club's membership of ACC.

Details of dates of meetings and further information can be obtained from John Schofield on Blackburn 28127.

ADD-ON EXTRA

Screen dump

Amstrad
£17.95
SMC Supplies
Suitable for Epson, Brother and Star printer owners, this pack contains a printer cable and a short program for dumping Amstrad screen displays. Colours on the screen are represented by different densities of shading.

Parallel interface

Commodore
£19.95
SMC Supplies
Cable and software, loaded in behind the kernel Rom, gives the CBM-64 a Centronics interface. Instructions are included for using the software with Easy Script and machine-code monitors.

Marconi

BBC/MSX/Commodore
£60
Tracker Ball
Until now only real men who liked dressing up like Michael Heseltine were allowed to play Missile Command with Marconi track balls. That tended to be slightly expensive — not to say dangerous with real missiles — so now Marconi is launching into the home this expensive joystick which will only come into its own with computer-aided design software.

E Interface

Spectrum
£39.95
Centronics Interface
Compatible with just about anything you can stick on a Spectrum, this new interface from Kempston comes with a metre of ribbon cable and is styled like the new Speccy +. The software in Eprom supports LPrint, LList and Copy commands.

Floppy Wise

BBC
£29.95
Disc Utilities
This Rom is compatible with the Acorn DFS and adds 14 new commands. One of these commands "clone" claims to copy a fully protected disc in under four minutes.

HARDWARE hotlist

Demon modem

- *BBC*
- *Demon Electronics Ltd*
- *£60 by mail order*

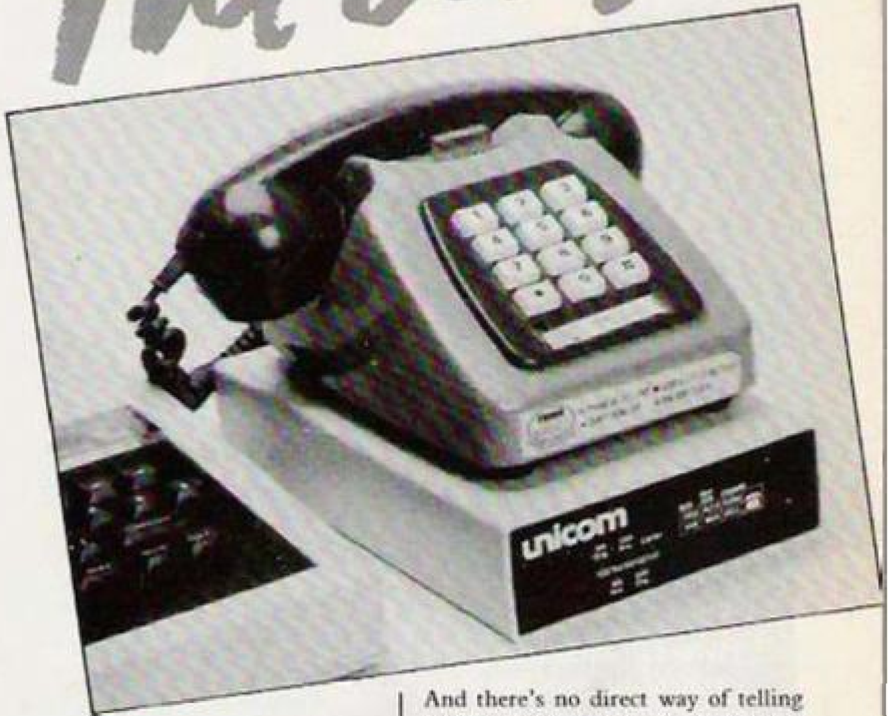
A MULTI-STANDARD modem for only £60? It's an astonishing price when other modems with the same sort of specification cost over twice as much. So does the Demon live up to its promise? Well . . . yes and no.

To avoid the high cost of switches, control of the Demon is delegated to your computer. Selection of signalling standard, switching on and off-line and so on is all done through the serial port, which also acts as a data highway. So it's essential to have the accompanying software.

At present the software is available only as a sideways Rom for the BBC Micro and it costs another £24. Apparently, it's still being finalised — which won't amuse those who sent off money when the Demon — *né* Unicom — was announced last autumn.

But the Rom is cleverly written and it certainly works hard. Everything is done by "star" commands: for example, *baud 1 selects the Prestel mode and *dial 618, for whatever, auto-dials the call for you.

It's easy enough to build the commands into your own programs. So with a bit of straightforward Basic you can create a bulletin-board which auto-answers the telephone, selects the right baud rate automatically and can even upload or download files.



The system is undoubtedly very flexible, but it might be awkward for the beginner to master. The Demon has no extension telephone socket, so you can't check for engaged tones or crossed lines unless you spend another £5 on a two-way adapter. And there's no direct way of telling whether your modem is holding the line.

One other snag is software clashes. The Rom is said to use legal calls only, but certain packages — such as Addcom and Computer Concepts' Graphics Rom — fight it.

But if you enjoy experimenting, the Demon will give you a lot of fun. Beware, though — it hasn't yet received BABT approval.

Disc drive

- *Amstrad*
- *£199.95*

IF YOU'RE SERIOUSLY into computing, then sooner or later you're going to need a disc drive. The flexibility and speed are so great that once you've used one, you'll wonder how you ever got along without it.

Amstrad's drive is further enhanced by the way it integrates well into the system as a whole. The drive consists of an interface, which plugs into the expansion socket in the back, a ribbon cable with two plugs, and the drive itself, which is a 3in. Hitachi unit.

When the computer is powered up, all normal tape commands control the disc. AMSDOS takes 1,280 bytes for work space, so some programs may not run with the disc installed. Apart from the tape commands, there are also several new disc commands which take the form of Basic extensions, using the "|" character as usual. Further Basic extensions can still be used normally.

Other utilities take the form of CP/M files on the supplied system disc: format, back-up and many more.



Obviously using the tape commands to control the disc has one major advantage: programs will need little or no alteration to run on disc; file names may need to be changed, AMSDOS is fussy about repeated or null file names.

The disc format is 3in. with 40 tracks, giving 180K per side. At £5 per disc, it gives fewer bytes for your £ than some systems, but if the system catches on, this price may fall.

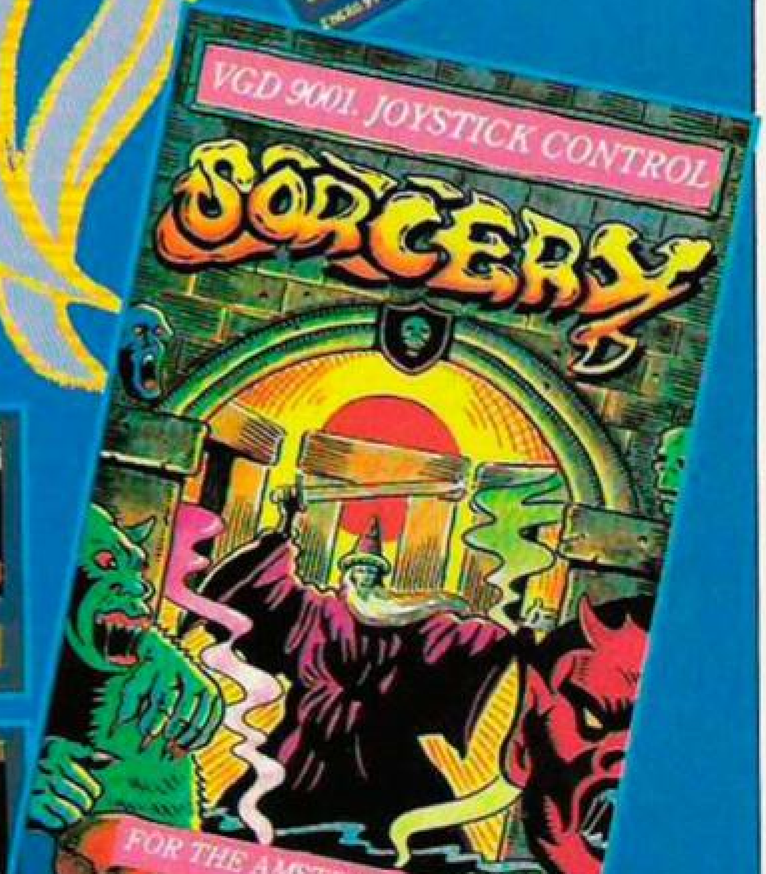
The system disc supplied has CP/M and the disc utilities, and a few demonstration files on one side and Dr Logo, the graphics language, on the other. Dr Logo is a fair implementation of the language, although it did seem rather slow.

The 100-page manual is clear and concise, giving all the information required on file types, transferring files from tape to disc, and giving brief notes on CP/M and the Logo language. Obviously if you were particularly interested in the last two, you'd need to buy a book.

A second drive can be added, of any type. Amstrad plans to bring out one at £169.95, but for the moment full information is available for DIY enthusiasts.

A good, no nonsense, well thought-out product, at a fair price.





VGD 9001. JOYSTICK CONTROL
SORCERY
FOR THE AMSTRAD CPC 464



Star★ chart

Chopper Squad
CPC-464
£6.00
Lunajetmanesque
Interceptor
★★

Nothing to do with the Aussie whirlybird TV series, you take the part of Macho, an ace helicopter pilot etc, etc. In the style of Lunar Jet-Man you have to assemble fighter planes by collecting parts and taking them off the bottom right-hand side of the screen, without hitting UFOs. I thought it was a terrible game at first, but it grew on me.

Aztec Tomb Revisited
CBM-64
£7.95
Alligata
★★

This is a graphic adventure with a very small graphics area in one corner of the screen. The idea is to find the other three quarters of the treasure map — you already have one quarter.

Obviously a follow-up to Alligata's earlier Aztec Tomb, they seem to have compromised on graphics in order to keep the over 40-command vocabulary. I think this will appeal more to text adventurers of the old school, rather than to Hobbit fans.

Wriggler
ZX Spectrum
£4.95
Arcade Maze
Romantic Robot
★★

Restful little game which comes with musical soundtrack "The moons of Jupiter" by Alex Goldscheider on the flip-side. In the 30 year-long maggot race you must move through four main areas: the garden, the scrubland, the underground and the mansion. You collect food and objects in the usual way. The graphics are very nice and this may appeal to people who find Atic-Atac too hectic.

SOFTWARE Shortlist

Alien 8
- *Spectrum*
- *Ultimate Play the Game*
- *Arcade adventure*
- *£9.95*

★★★★

FORGET the superficial lost-in-space flavour of the packaging, this game's more Castel Gondolfo than spaceship Nostromo. Not so much Alien 8 as Knight's Lore 2.

That said, Knight's Lore is the definitive arcade adventure and Alien 8 retains the chunky three-dimensional Filmation graphics which distinguished the original. Again you have to combine manual dexterity and a quick mind to solve the problems posed by each room before you can pick up a treasure or progress to the next chamber of the maze.

If you want to approach the game unaided stop reading now — and no sneaking a look at the maze map.

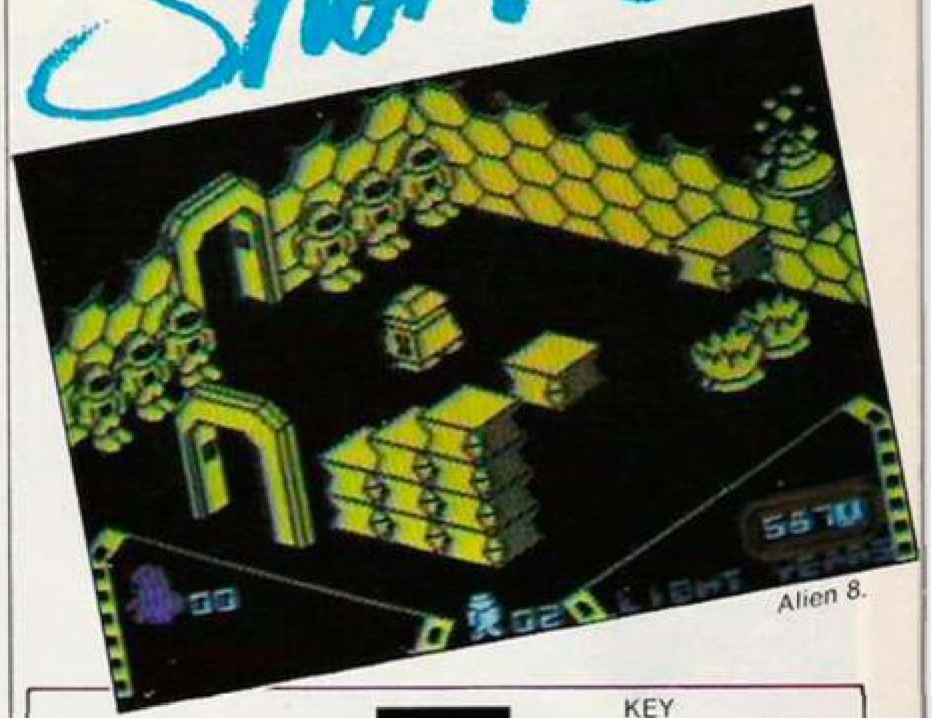
The plot — such as it is — derives loosely from the film. You control a robot padding around a craft carrying cargoes of deep-frozen cryonauts. Unfortunately assorted extra-terrestrials — clockwork mice, rodent daleks and the like — have unplugged the refrigeration.

To prevent your cryonaughts looking like a freezer full of fish fingers after a power cut you have to collect power packs shaped like pyramids, blocks, mushrooms and cheeses and plug them in to similarly shaped flashing sockets.

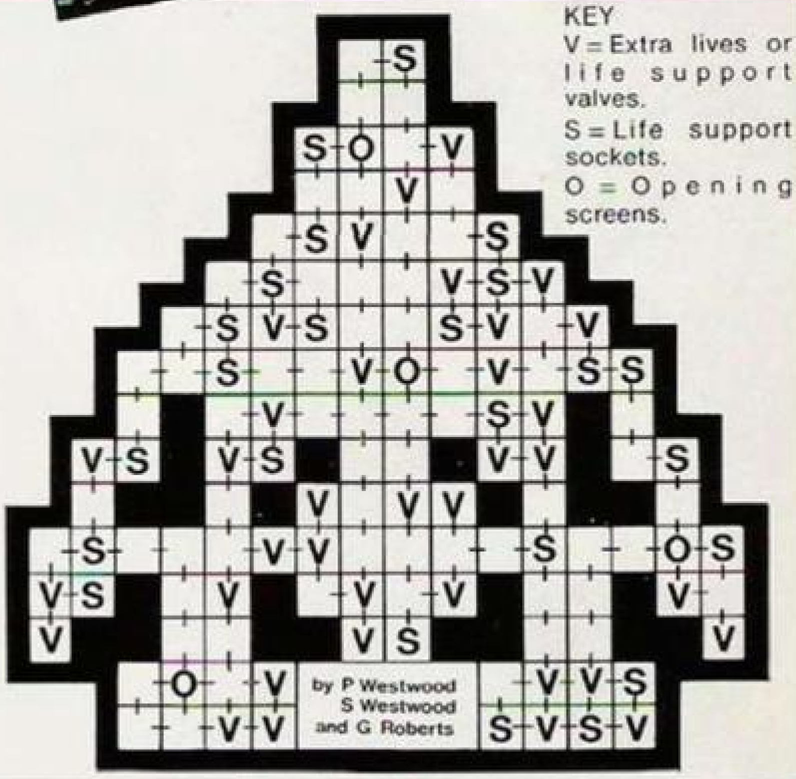
You can also use the packs as bulldozers to clear minefields or steps to help you jump higher — holding the jump key down can also help. Some of the steps vanish or sink — others are booby-trapped with an escalator effect which drifts you towards the nearest danger.

Perhaps the neatest new trick in Alien 8 is a 3D cursor pad on screen with which your android can direct another remote-controlled robot to clear a safe path.

Meirion Jones



Alien 8.



Below: Everyone's a Wally.



Everyone's a Wally
- *Spectrum 48K*
- *Mikro-Gen*
- *Arcade adventure*
- *£9.95*

★★★

"EVERYONE'S A wally, a charlie or a bungler or a berk" warbles Mike Berry on the "hit single" recorded on the flipside of this cassette. The song will drive you bonkers marginally quicker than actually playing the game which, it must be said, has excellent graphics and demands great perspicacity.

(continued on page 47)



Multiple Combat Scenarios
Full Aerobatics
3-D Graphics
F-15 STRIKE EAGLE
U.S. GOLD
All American Software
MicroProse Software
64
ATARI
F-15 Strike Eagle features superb 3-D graphics, F-15 manoeuvreability, computer assisted targetting including modern fighter Heads Up Display, airborne radar, air to air missiles, surface to air missile defence, ground target bombing, full aero-batics, and multiple combat scenarios.
DISK/CASSETTE £14.95
Solo FLIGHT
SPITFIRE ACE
NATO COMMANDER
SOLO FLIGHT: Take off with this user friendly flight simulator which includes 3-D graphics, full cockpit instrumentation, dual VORs, ILS, 21 different airports, and real life challenges from weather and mechanical emergencies. DISK/CASSETTE £14.95
SPITFIRE ACE: Save London during the Blitz in your RAF Spitfire! Outstanding out-of-the-cockpit 3-D graphics; 14 different combat senarios. DISK £12.95, CASSETTE £9.95
NATO COMMANDER: Defend Europe in an accelerated real-time war situation! Full scrolling battle map, includes infantry, armour, helicopters, airforces and tactical nuclear weapons. DISK £12.95, CASSETTE £9.95
U.S. Gold is stocked by all leading computer stores including:
Boots
John Menzies
WHSMITH
WILDINGS
WOOLWORTH
U.S. Gold Limited, Unit 10, The Parkway Industrial Centre, Heneage Street, Birmingham B7 4LY.
Telephone: 021-359 3020. Telex: 337268.

Star chart

Halaga

ZX Spectrum
£5.50
Zalaga clone
Interceptor
★★

If you've never played Galaga in a bikers' roadhouse in Winnipeg, you've probably had the opportunity to play Zalaga on the BBC Micro. Now you can play Halaga on the Spectrum. Have to be a bit careful about which keys you press — this one shows a tendency to lock up. If you can keep it going, this is a great traditional shoot'em-up.

Alpha-Beth

ZX Spectrum
£5.75
Educational
A 'n' F Software
★

The sort of education game that says "Your not trying" when it means "You're not trying"., Don't you just hate teachers who are stupider than you are? The program assures us that Indira Gandhi is the prime minister of India. Sorry, A 'n' F, some good ideas but must try harder.

Hyper Circuit

CBM 64
£7.95
Shoot'em-up
Alligata
★★

Good graphics, but endless blasting away seems more pointless than usual. From the aerial view you seem to have been catapulted into some nightmare future city where all the trains have been privatised and are busily tooling around between city blocks shooting each other.

Snake Kink

ZX-81
No man's land
Eat-'em-up
£3.99
★★

Written by Laurent Mazo this French offering is nice and simple. Controlling your snake by 9 and 0, you must eat 10 apples and pass through the door to finish every level. If you touch your own body or slide into the wall you lose a life. Nostalgic.

(continued from page 45)

Mikro-Gen, with games of the quality of Pyjamarama, have carved themselves a niche in the market that is as distinctive in its way as that of Ultimate Play The Game. Both companies produce games that may have clear goals for players to achieve but whose appeal lies in leaving it to the player to deduce methods of attaining these goals.

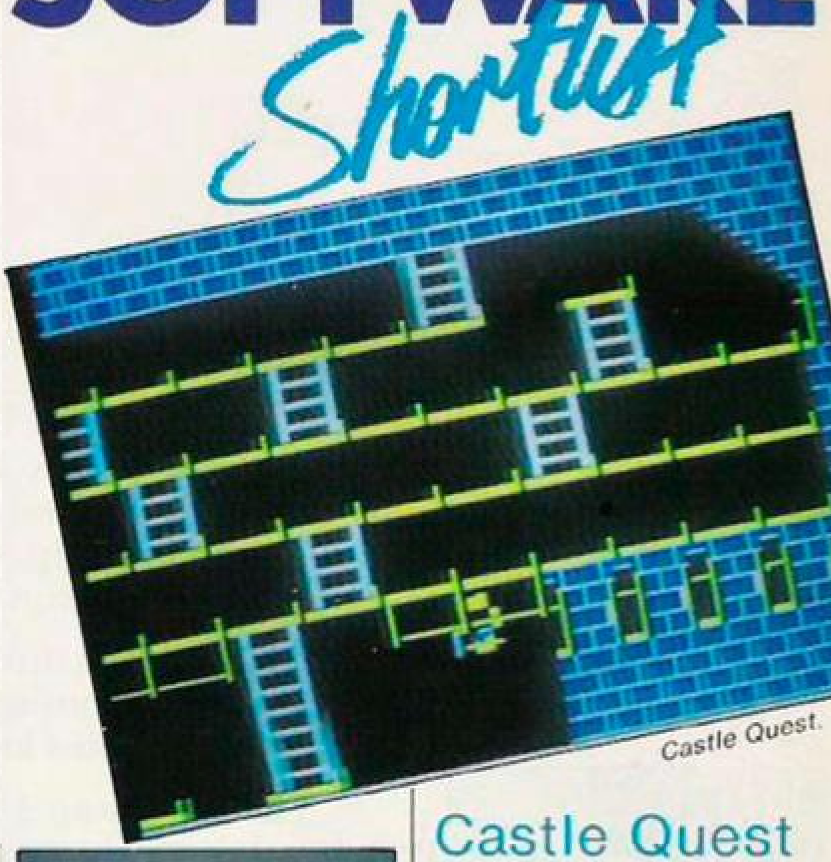
The complexity of this game lies in the facility to transfer control from the central character, Wally, to other members of the gang, Wilma his wife, Dick the plumber, Tom the punk mechanic, and Harry the hippy electrician. There is also a baby called Herbert who just trips people up.

The aim of the game is to open the safe in the bank, to pay the gang's wages. The combination of the safe is discovered in the process of performing various tasks. Different members of the gang excel at different tasks. Male chauvinist pigs will be pleased to know that Wilma is best at shopping, for example. But her real job is to get three library books back to the library.

Each character's endurance level has to be maintained in the traditional way by eating food.

Paul Bond

SOFTWARE Shortlist



Castle Quest.

Superchess 3.5

- *Spectrum 48K*
- *Deep Thought — CP Software*
- *Board-game*
- *£9.95*

If you're the sort of chess player who needs to know exactly what's going on in your opponent's mind, this is the program for you. If you also want an extremely fast, challenging game, the 1985 version of Superchess again fits the bill.

Down the left side of the board display is Plymax. It not only shows you how far ahead the computer is looking, but also details its best line of play together with your answering moves. Beginners will find this useful because it draws attention to traps and developments they might otherwise have overlooked.

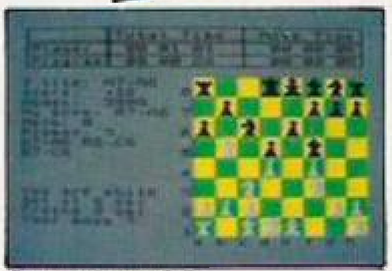
The screen also displays the times the computer and the player have taken in all, as well as the time taken for the current move. As the program is working out what to do next it tells you the move it likes best so far, how many nodes or lines of play it has analysed and your last move.

Unlike most chess programs the pieces are moved round the board not with the cursor keys but by coordinates — for example, E2-E4.

The way you set the level of play is again unconventional. With Superchess the amount of time you are prepared to give the computer to think over its moves determines the quality of the game. Obviously the more time you give it the better it will play.

Even with only an average of 10 seconds response time the program can give you a very good game.

Toby Wolpe



Superchess 3.5.

Hellfire

- *Spectrum 48K*
- *Melbourne House*
- *Arcade Adventure*
- *£6.95*

★ ★ ★ ★

Those whom the gods wish to destroy, they first make mad. Melbourne House's part in this universal scheme of things is to produce "Hellfire", a game which purports to be based on the trials of Ulysses.

We start out with hill trials where, stabbing deftly at the Q,S,A etc keys you help the hardpressed Greek hero negotiate the slopes of Olympus. Falling boulders conspire to reduce him to the consistency of a plate of squashed moussaka — then he has to duck the stony stare of the repulsive Gorgon. After this he finds himself in the temple of Knossos — nowadays beset no doubt by minor tourists, but then plagued by minotaurs — almost as bad, but not nearly as contrived as this last pun.

Strong graphics, comparable with those in the definitive Tir Na Nog, but this program is rather more arcade than adventure, if you follow my drift.

On the final screen you must locate the exit and use your mace to kill monsters and break exists open. Extra maces can be found in the chests scattered throughout the maze.

Paul Bond

Castle Quest

- *BBC*
- *Arcade Adventure*
- *Micropower*
- *£12.95*

★★★★

They bet you a pound you can't crack it and they are likely to be right. Castle Quest presents the player with some fiendishly difficult problems to solve within the first few screens of play. It is, in fact, one of the few arcade adventures where you need the real adventure mentality to get you anywhere.

For instance, how do you escape after a patrol of green guards has frog-marched you to the dungeons? Well, we can't give too much away, but the answer lies somewhere between a torch, a stool, and a bed. And that spider that always gets you as you go through the entrance to another area? Those funny little monkeys running around just might be able to give you some help.

Text adventure fans will feel instantly at home here, but the game should also ensnare sceptics such as this reviewer, to whom the prospect of playing an adventure game normally has as much appeal as Hangman on the ZX-81. Being able to see an aqualung or a wand before you pick it up is rather more satisfactory than reading a bald description of it in print.

In addition to this the game offers excellent animation and sound together with some unusual scrolling effects. The castle itself is a multitiered assembly of ladders, tunnels, cakewalks, steps, platforms, and pinnacles; when you approach the edge of the screen it pans smoothly upwards, downwards or sideways to reveal the next section.

Simon Beesley

(continued on page 49)

The Dawn of a new beginning for your Spectrum

Introducing Discovery 1 – quite simply the most advanced and comprehensive disc drive unit yet developed for the Spectrum and Spectrum + computers.

Discovery is the breakthrough in design, performance and price which promises to make other Spectrum solutions obsolete. Just look at its advanced features:

- The very latest 3½ inch disc drive unit
- Peripheral through connector
- Parallel printer port
- Joystick interface
- Video monitor port
- On-board power supply



But the biggest discovery of all is the price – just £199.95

Upgrading to a dual-drive system is simple with the 'Discovery +' upgrade package – £139.95, or choose a purpose-built twin-drive unit, 'Discovery 2' – £329.95.

THE DISCOVERY RANGE
- Available exclusively through larger branches of Boots and supported by a brand new range of popular 3½ inch disc software.

AVAILABLE FREE
- Memorex disc twin-pack + five free discount vouchers.

Further product information is available from

Opus. Opus Supplies Ltd., 158 Camberwell Road, London SE5 0EE
01-701 8668, 01-703 6155

SOFTWARE Shortlist

Star★chart

Interdictor Pilot

Amstrad CPC 464
£17.95
Simulator
Supersoft
★★★★

Perhaps a touch on the price side but, if you want the best, this is it. For your money you get a sturdy plastic case, compendious instructions, hints and "background" and of course the tape. A conversion from the C64, reviewed last November, it features improved graphics, but is substantially the same. My only gripe is that there is no keyboard option and the joystick just isn't really accurate enough for the fine control required. Still the best space flight simulator by a long chalk.

Toy Bizarre

Spectrum 48K
£7.99
Platform
Activision
★★★

A follow up to their smash hit Ghostbusters this game treads once more the well worn path of the platform game. Is the market for these things completely insatiable? What we have is a series of screens, one much like the next. You have the mindless task of bursting balloons before they reach the ceiling and turn into the eponymous bizarre (and hence deadly?) toys. There are a few other bits and pieces, like platforms, valves and a homicidal maniac called Hilda, but what we have is essentially the same old stuff.

Aqua Racer

CBM-64
£6.99
Race game
Bubble Bus
★★

Pole Position on water. Screen display depicts a rear view of a powerboat. Steer down the course, red buoys to starboard, black and white to port. Avoid crashing into other powerboats. Use firebutton to chang gear. Choice of several courses. Addictive, needs fast reflexes.

(continued from page 47)

Sorcery

- *Amstrad CPC 464*
- *Arcade Adventure*
- *Virgin Games*
- *£8.95*

★★★★

Being a graphic designer with a software house used to be a pretty unrewarding way to earn a crust. Apart from doing a nice title screen which normally had nothing to do with the game and annoyed the punters like hell because it took ages to load, there was nothing left. They might let you design a few sprites, but as they normally ended up as a few fuzzy splodges on the screen, what was the point. Enter the Amstrad, answer to your prayers. All 40 screens and all the sprites in this game have clearly had hours of tender loving care lavished upon them. For those not merely content with watching the graphics, what's the game like?

As usual it's one man against the world. Your task, as the last free sorcerer, is to rescue your colleagues imprisoned by the evil necromancer. The objects you need to free them have been left rather carelessly strewn around the place. You can carry one object at a time, and it's just a question of sussing out which object unlocks which door.

Were life that simple. You also have to contend with various nasties which can be disposed of if you have the right object, or will dispose of you if you haven't.

Lee Paddon

Spider-man

- *ZX Spectrum*
- *Adventure International*
- *Graphic adventure*
- *£9.95*

★★★

Keep cool, pilgrim. Mary Jane, my ex, knows I'm really Peter Parker, my boss at the Daily Bugle doesn't dig the freelance photos I've been submitting, my aunt won't speak to me since I dropped out of graduate school, and my current girlfriend — the Black Cat — can't stand Peter Parker. Only likes me as Spider-man.

No-one knows better than Spider-man what a tangled webv we weave when first we practise to deceive. And so we enter the wacky twilight world of Marvel comics where the characters, with their secret identities and double lives, are cathartic reflections of the schizophrenia inherent in modern Western society *(What's going on? — Ed).*

Spider-man is the second adventure in the Questprobe series, a combination of comics and computer adventure games that allow you to become your favourite Marvel superhero.

The screen gives you a static display of the area you are standing in — watch out for piles of sand — they turn into Sandman.

Paul Bond



Sorcery.

Dark Star

- *Amstrad CPC 464*
- *Arcade*
- *Design Design*
- *£7.95*

★★★★

It's conversion time again. This time it's an old Spectrum game having the Amstrad experience. If you missed this one first time round, it's an arcade shoot-em-up with strategic overtones.

The galaxy is in the tyranical grip of a militaristic dictatorship. You have to go out there and liberate planets by the simple, if rather bloody, expedient of blowing up everything in site. Sounds familiar? Well, if not hot on originality, this game scores with execution. What it boils down to is a series of linked mini-scenarios, each of which consists of getting down onto the planet, finding the enemy base, dodging the flak and steering through the holes.

After one base has been dealt with, pausing only to tank up the old shields, it's off to the next planet and more of the same. All this might seem to pall a bit after a while, but the graphics and speed are breathtaking.

There is also a design-your-own "front end" to the game which allows you to change the game to suit your tastes.

This game is going to make as much impact with the Amstrad fraternity as it did with Spectrum owners.

Lee Paddon



Dark Star.



ACE.

Air Combat Emulator

- *C-16*
- *Cascade*
- *Combat emulator*
- *£9.95*

★★★

This is one of those "given the limitations of the machine" kind of reviews. ACE — I trust you to work out the acronym for yourselves — is a very fast, punchy flight simulator — except that everyone from Farnborough Air Show commentator Raymond Baxter to Nigel Stevens and Guy Wilhelmy of Cascade was very keen to say "emulator" rather than "simulator" at the launch demo.

The reason is you don't get to take off and land like you do on Fighter Pilot or Strike Eagle. You're airborne and if you want to stay that way you have to shoot down enemy aircraft.

The screen display shows how many rockets you have, about every fourth one is a smart missile and homes on its targets. The real trick to this game is co-ordinating the roll and pitch aircraft-symbol indicators so that you don't overdo any climbing or diving. Targets appear as upward arrows on the radar if they are above you; downward arrows if below. You have to get them to be squares which means they are at the same altitude as you, though even then you may not necessarily be able to see them.

Paul Bond

(continued on page 51)



QUALITY NEVER LOOKED
SO GOOD–
Now everyone with a BBC or IBM PC will want to get their paws on CUB's sleek new D series plastic cabinet – a triumph of ergonomics and up to the minute design. Within it is the CUB 653 MEDIUM RESOLUTION colour monitor – the perfect mate for computer users who wish to combine the advantages of brilliant, low cost colour graphics with 80 column processing software.
CUB 653's remarkable depth of colour is enhanced by minimal screen glare, thanks to a super high contrast CRT. Even in well lit environments the 653 (H) × 585 (V) resolution and 0.43mm dot pitch produces 80 column text which is pin-sharp and easy to read. Owners of SHARP, RML 480Z, APPLE Series, WANG and other leading computers needn't feel left out, because CUB 653's compatibility extends to these models and many more.
PICK ONE UP
FOR JUST £299 inc. VAT
NEW LOW PRICE,
NEW PLASTIC CABINET –
THERE'S NEVER BEEN A
BETTER TIME TO BUY.
Standard Resolution version also available in new cabinet. Both Standard and Medium resolution models produced in metal cabinets if required.
MICROVITEC
CUB 653
COLOUR DISPLAYS
Microvitec PLC, Futures Way, Bolling Road, Bradford, BD4 7TU, West Yorkshire.
Tel: (0274) 390011 Telex: 517717
Available from High Street Computer Retailers and selected branches of W. H. Smith, Harrods, John Lewis Partnership, John Menzies with selected models available from larger branches of Boots.

Star★chart

Upper Gumtree

CBM-64
Richard Shepherd Software
£6.50
Graphic adventure
★★★

Nice graphics, a touch of humour — all the makings of the second wind that Richard Shepherd Software need to keep abreast in what is supposed to be a very competitive market. Can you save the world from the evil professor Blowtovitz? Rusting Ford Anglias, cows lying on their backs in purple fields — give the public what they want.

Flight Simulation

Amstrad CPC 464
Myrddin
£11.95
Flight Simulation
★★★

Myrddin can count themselves unlucky that this game has come out at the same time as Digital's Fighter Pilot. While it is a relatively competent sort of package, it doesn't compare well with its illustrious competition. You stooge around the sky, with rather wooden controls. There are a few bits of ground detail and 15 different models of plane to fly. No ILS makes landing a bit tricky and, of course, there's nothing to shoot at.

747

CBM-64
Docsoft
£12.95
Flight Simulation
★★★

What do you get when you confront an experienced pilot but inexperienced programmer with a C64? Probably something rather like 747. The manual looks very promising with instructions on approach, holding patterns, navigation and the like. However, not all is adequately explained or, if it is, not for your average greenhorn. The disappointment starts when you load up the game. Ever tried aerobatics with a Jumbo, now's your chance.

(continued from page 49)

Wizard's Lair

- *ZX Spectrum*
- *Bubble Bus*
- *Arcade adventure*
- *£6.99*

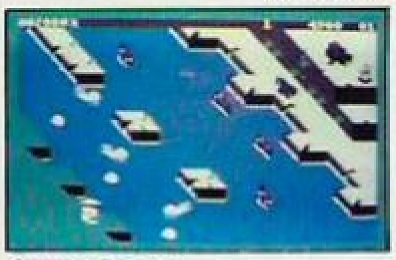
★★★

Wizard's Lair by Stephen Crow is a turn-up for the books as far as Bubble Bus is concerned. It should appeal to fans of Atic Atac, which is a roundabout way of saying that what it lacks in originality of inspiration, it makes up for in excellence of implementation.

The situation is certainly pretty hopeless inside, beset as you are by the usual nauseating bunch of dragons and strange blobs and if you're really unlucky a huge purple or occasionally green cut-out serpent.

Objects to be collected include treasure chests, but what you are really trying to accumulate is pieces of the golden lion. There are five of these just lion around all over the place. There are also keys, diamonds and rings.

Paul Bond



Congo Bongo.

Congo Bongo

- *CBM-64*
- *US Gold*
- *3D Kong Frogger*
- *£9.95*

★★★★

Basically a two-screen game, each screen has several different levels of difficulty. And although the basic inspiration behind the first game is clearly drawn from Donkey Kong and that of the second screen from Frogger, both ideas have been very artistically re-implemented.

On screen one you have to scale Jungle Mountain while the insensate simian rolls what look like purple rugby footballs at you.

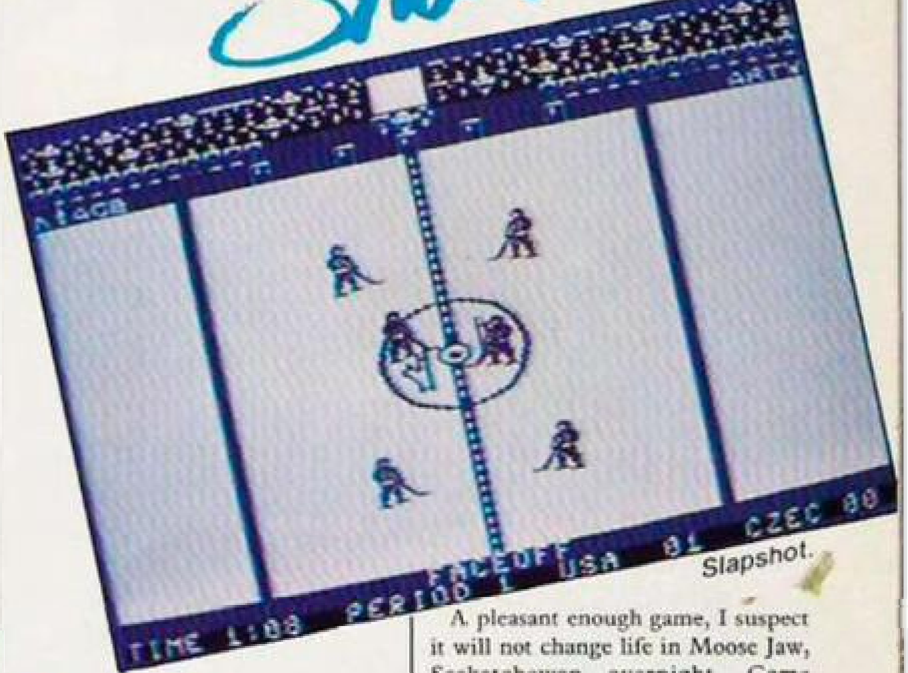
The action takes place on a set created out of giant building blocks which you hop around on, hopefully not falling into the water, falling off a ledge or succumbing to a forward pass from the purple footballs. When you jump across the chasm, the cliff falls and you can't get back.

There are a lot of monkeys dodging about, but they don't bother you at the first level of play. As you get smarter, more of them start to interfere with your jungle jaunt.

When you reach Congo's perch you automatically move to the next screen. This is a river chock-full of the usual old hippos, pink alligators and lily pads. There is also a blue rhino rampaging systematically about on the far bank.

Paul Bond

SOFTWARE *Shortlist*



Slapshot.

Slapshot

- *CBM-64*
- *Anirog*
- *Sport simulation*
- *£8.95*

★★★

"He scores!" cries the computer in a way that is instantly reminiscent of Ghostbusters' "He slimed me!" This two-player ice hockey program will inevitably be compared with the excellent International Soccer game — although with Commodore's Canadian connections, ice hockey might be a natural for this machine.

A pleasant enough game, I suspect it will not change life in Moose Jaw, Saskatchewan overnight. Come winter, Pop will still flood the backyard so the kids can slither around on the ice, batting hell out of each other.

Teams are drawn from the six that competed in the 1984 Canada Cup series: Sweden, Czechoslovakia, West Germany, Canada, the Soviet Union and the USA. The player nearest to the puck is the one that the joystick controls, as in International Soccer. When you pass, control automatically moves to the player of your team who is in closest proximity to the puck, which even casts a shadow if you hit it hard enough to make it take off from the ice.

Paul Bond.

Shades

- *CBM-64*
- *Durell*
- *Arcade Adventure*
- *£6.99*

★★★★

From the people that brought you Harrier Attack, Scuba Diver and Combat Lynx comes this foray into the world of the supernatural. The shades of the title are not hip sunglasses but unpleasant wraiths which, along with other denizens of the nether world, impede you in your quest to rid the land of said shades. The curse of the shades is that they cause people to disappear from one location and reappear somewhere they didn't expect to be. Anyone who has travelled extensively by British Rail will be able to identify with this.

Moving, fighting and pleading are options open to you in this game, as well as in your use of public transport. The joystick directs you from one location to another. As you journey on you encounter various monsters, creatures and magical objects. The objects take the form of brown chests, and you can carry a maximum of eight.

Paul Bond



Gryphon.

Gryphon

- *CBM-64*
- *Quicksilva*
- *Defenderesque*
- *£7.99*

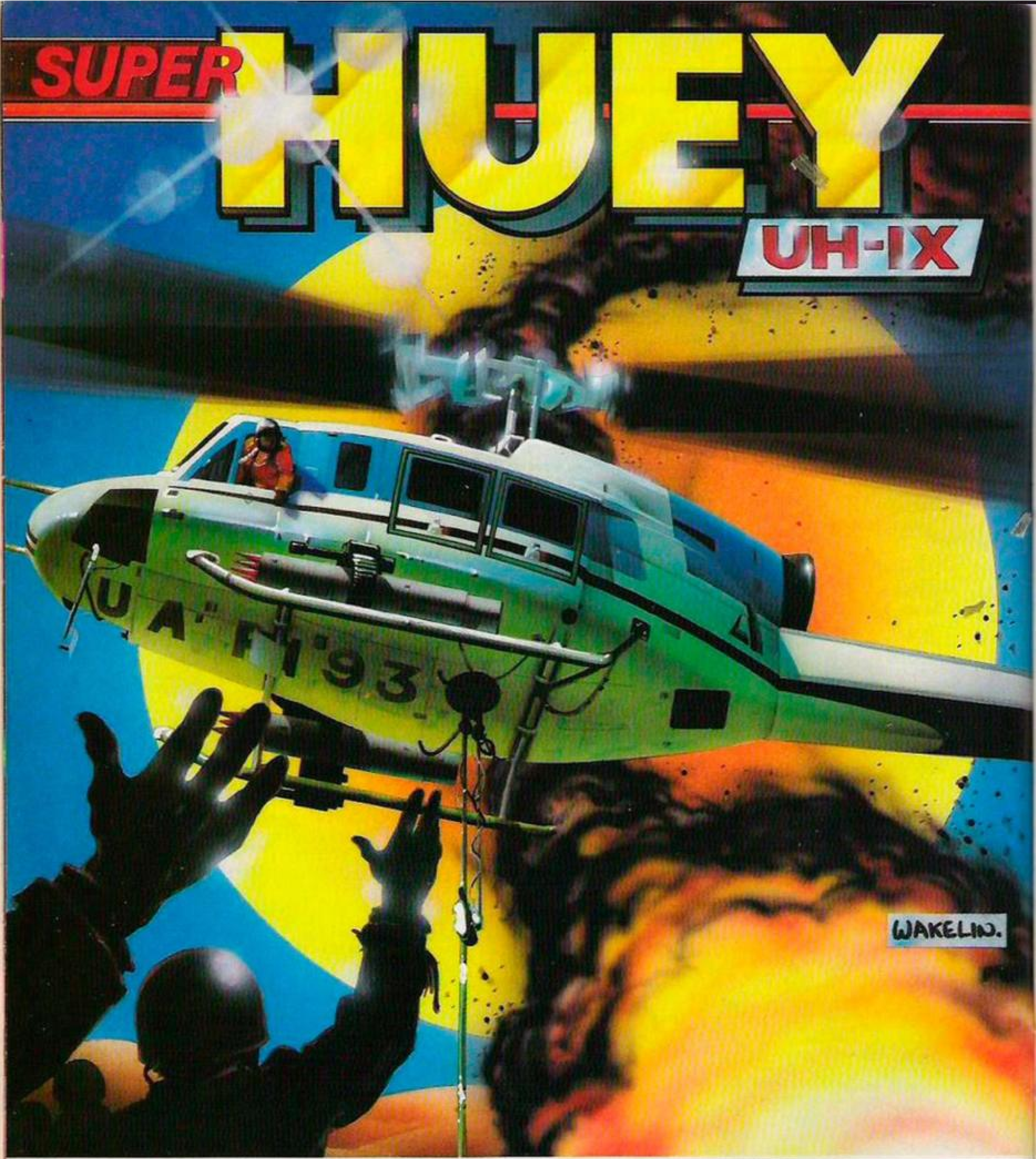
★★★

Beautiful blue 3D Disneyesque scrolling graphics like that bit in Bambi when they're in the wood — sorry, let's not get too carried away.

Billed as an arcade fairy tale, the main protagonist is a delightful furry winged animal whose gold horde is threatened by Id Monsters. These are inoffensive-looking little white ghosts which you disperse in the customary way by firing bolts of pure Gryphon magic at them.

You use your gold bars to make stepping stones through the deadly waters which cover the floor of the tunnel.

Paul Bond



SOLO FLIGHT – Computer controlled flight training teaches the techniques of flying and familiarization with instruments.

RESCUE – Your assignment is to locate and attempt pick up of stranded military personnel.

EXPLORE – Fly a survey mission over previously unexplored territory and map the terrain.

COMBAT – Use your rockets and machine guns to defend a secret desert installation from unknown hostile forces.

HELICOPTER FLIGHT SIMULATOR

U.S. GOLD
All American Software

COSMI

CASSETTE **£11.95** DISK **£14.95**

64

U.S. Gold Ltd., Unit 10, The Parkway Industrial Centre, Heneage Street, Birmingham B7 4LY. Tel: 021-359 3020.

ATARI

QUEST Corner

NEWS
oaeo
rssu
tttt
h h

The crystal ball

Spectrum owners now have the opportunity to play a Commodore 64 favourite, Heroes of Karn. It doesn't have as many graphics as the 64 version but the quality of those that have been included are far, far superior. Credit for them belongs to Terry Greer who also designed those for Interceptor's Jewels of Babylon.

Commodore owners have not been forgotten. Interceptor have also released a sequel, Empire of Karn.

Two new adventure-solution books, £3.95 each, have been published by Duckworths. The Spectrum Adventurer provides detailed solutions to Snowball, Twin Kingdom Valley, Urban Upstart and Valhalla.

The Commodore 64 Adventurer offers solutions to Heroes of Karn, Lords of Time, The Count and Voodoo Castle.

A helping hand

Andrew Parker of Warrington has run up against a few obstacles in Infocom's ZORK II. For Andrew and other Zorkers, here's a few titbits:

Can't get past the lizard head?
YDNAC FO EGAKCAP TI EVIG

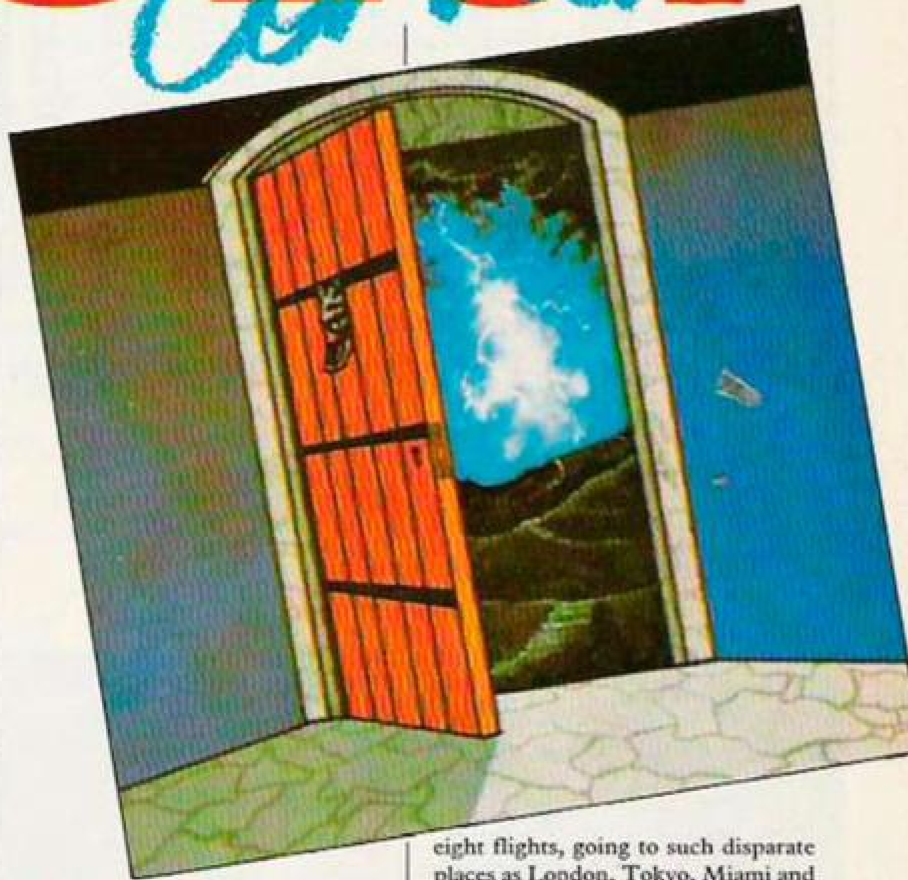
Red sphere proving elusive?
EGAC TFIL OT TOBOR LLET NEHT IT TEG

Prevented from getting the hazy object in the pool of tears.
EKAC DER EHT NI GNIWORHT YB LOOP PU YRD

Attention Snowball fans. Help is at hand:
Don't know how to mend the damaged droid?
(a) ODNOC ENIMAXE
(b) SDEL TEG
(c) DIORD OT EBORP HCATTA

Yours in wells and woes,
HUGO NORTH

Our man with the brass lamp and the key to a thousand mysteries sheds light on new adventure programs. Lost? Never fear, Hugo North is here.



Staff of Zaranol

48K Spectrum
£5.95
CCS (Cases Computer Simulations)

IN THIS Quilled text adventure, you play the part of a junior wizard. While your mentor is away attending the Sorcerer's annual binge, you meddle with his spells and inadvertently conjure up a demon. All your efforts at banishing the creature fail. Happily, the demon decides of his own accord to push off — less happily, he curses you as he evaporates.

You've got it all to do now. Here's the shopping list: discover the demon's name, find out the nature of the curse, seek out the Master's secret room, and finally banish the demon by using a dragon's tooth and the Staff of Zaranol. Tall order, eh? Serves you right for messing around with the sorcerer's spell books.

As if you hadn't enough woes, although your brain is telling you to get cracking, your legs have turned to lead. There you are in your own room, an open door and an interesting chest beckoning, yet your walking gear has gone on strike. At this point you'll probably be wishing you'd taken up banking, instead of wizardry, as a career.

Don't despair, help is not far away. A necromancer called Lan will appear whenever you call. Try summoning him to open the chest. He will do so, remove a knife from therein and advance towards your paralysed self . . . You begin to think this is going to be one of those days.

The adventure is atmospheric, there are plenty of puzzles and objects to savour and the plot is engrossing. Staff of Zaranol should keep you enjoyably engaged.

Amazon

Commodore 64
£19.95
Trillium Software

INTERACTIVE FICTION is the label applied to this text and graphics adventure. The game is not cheap but it's big, coming on two double-sided discs. You get a large adventure for your money.

As I sat before the console in the communications room of a top-secret research company, a transmission started to come through. On the screen, pictures appeared from the Amazon field team — but something was horribly wrong. Those were bodies strewn around the camp. Before my startled eyes a menacing, war-painted face loomed into view and filled the screen. Then the transmission broke up.

The field team were on assignment to find a source of DV emeralds. No ordinary emeralds, these contain microscopic platinum impurities (essential for use in hush-hush projects, naturally). My mission was to follow in the footsteps of the slain Amazon team and go get them gems.

Following the transmission, I was summoned to Director Murphy's office for a briefing. He's not a man to mess around with; one cheeky answer and Murphy slung me out of his office — and the game!

I later received a sealed envelope and was told to head for the airport. But which plane to take? There were eight flights, going to such disparate places as London, Tokyo, Miami and Cuzco. I chose Miami but had barely touched down before being mugged by a cabbie!

Amazon was written by Michael Crichton, author of such popular novels as The Andromeda Strain and The Terminal Man. The slowness of the Commodore 1541 disc drive is a drag but if you don't mind that, then you should get a lot of pleasure playing Amazon.

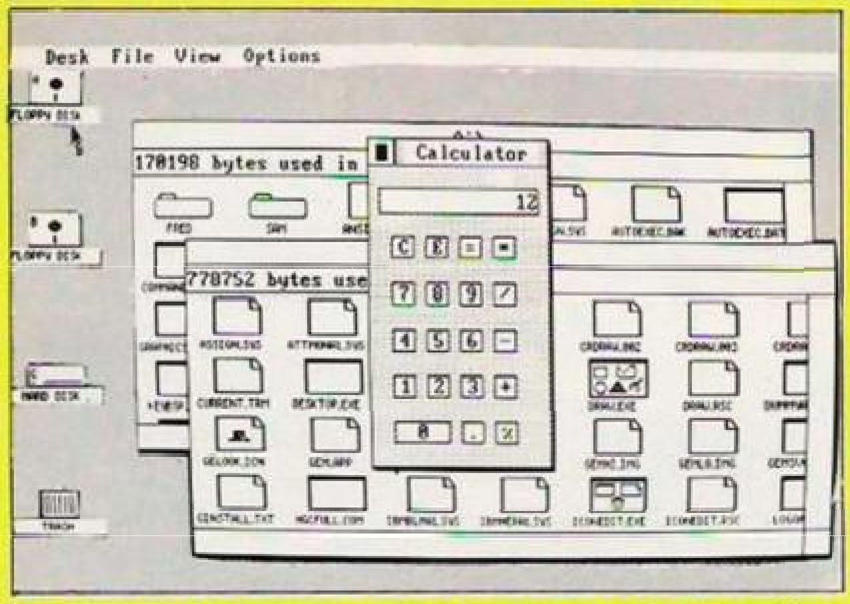
The Sandman Cometh

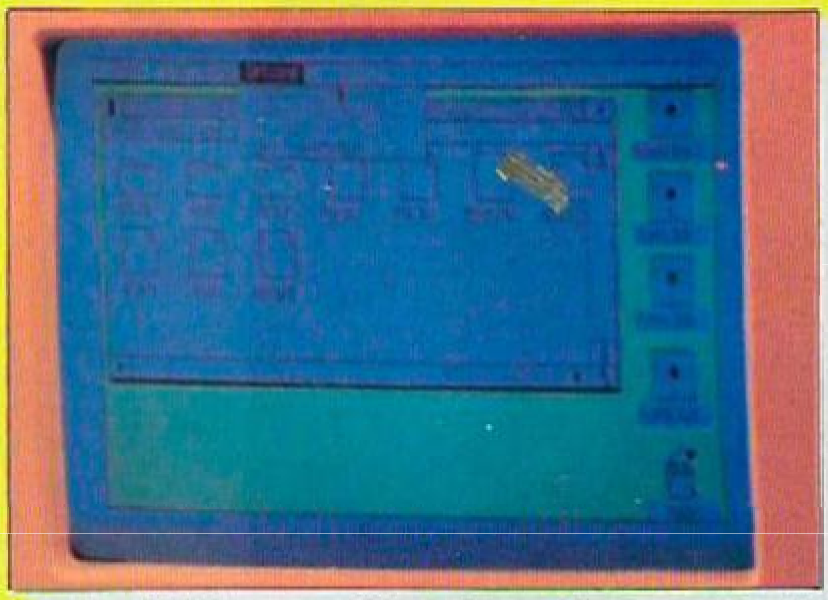
48K Spectrum
£10.95
Star Dreams

THIS TEXT and graphics adventure has a novel theme — it takes you through a sequence of dreams, the puzzles and clues getting tougher as you prgress. It comes in two parts, one on each side of the cassette.

I began by walking down some stairs to a locked door. Having found a brass key, I could only get past the door with the unusual command combination of Use Brass — "Use Key" didn't work — then Through Door. It wasn't long before I was exploring a fairground, being challenged by a gunslinger, locked in a cell and confronted by the Cheddar — no, not Cheshire — cat.

Quite enjoyable, even though some of the commands are strange. At least a Vocab command supplies you with a list of verbs while the Help command sometimes does just that.





Call Jack Tramiel a wimp at your peril — but call his new ST micro that and he'll be delighted — he thinks these machines will change home computing.

ATARI'S SPRAT TO

"HOME COMPUTER?" — big-talking Jack Tramiel gathers his bulldog features into an incredulous smile as if his questioner was mad even to mention such a thing — "I never heard of it — I make personal computers". If Tramiel's Atari has its way 1985 will be the last that any of us hears of home computers.

Instead, Tramiel looks to bring the power of business micros and the user-friendliness of machines like the Apple Macintosh into the sub-£400 price bracket for the first time, with his new ST range. Simultaneously, Atari will make their traditional 8-bit "home" micros more serious by packaging them with useful software or peripherals like the aggressively-priced current offer of a 64K Atari 800 XL with a disc drive and a filing program for £250.

So while the rest of the micro industry wobbles — Commodore announces falling annual sales figures for the first time, Sinclair has to delay his stock market flotation, Acorn needs to be rescued by Olivetti and smaller companies like Oric fall into the hands of the receivers — Tramiel proudly boasts that he will produce a record five million Atari computers this year confident that the market is there for the right product at the right price.

So far Tramiel has been able to shrug off the doubts expressed by his rivals, that he will not be able to produce a machine with the promised performance without being late or having to hike the price up. When asked about sceptical comments on the new Ataris by Sir Clive Sinclair he replies with a "Who?" but so far Tramiel has only been able to show us a prototype — exciting but incomplete.

What makes the ST computers so different from today's home micros is its use of Digital Research's Graphics Environment Manager, Gem for short, to soften the barrier between the user and the machine which Tramiel thinks is a major reason for "technofear". "I have a very hard time typing with two fingers — a mouse is easier — I believe there are a lot of people like me".

The mouse, which plugs into the ST's joy-stick port, is all part of the window, icon, mouse programs which give systems like Gem the unfortunate generic acronym Wimp. Wimps first appeared in production on Apple's £6,000 Lisa micros in 1981 before graduating to the £2,000 Macintosh in 1983. But as Tramiel points out, the ideas originated in Xerox's Palo Alto laboratory 10 years ago, where research was concentrating on doing away with the keyboard to make computers more friendly. "Xerox developed it, Apple copied it, I improved it".

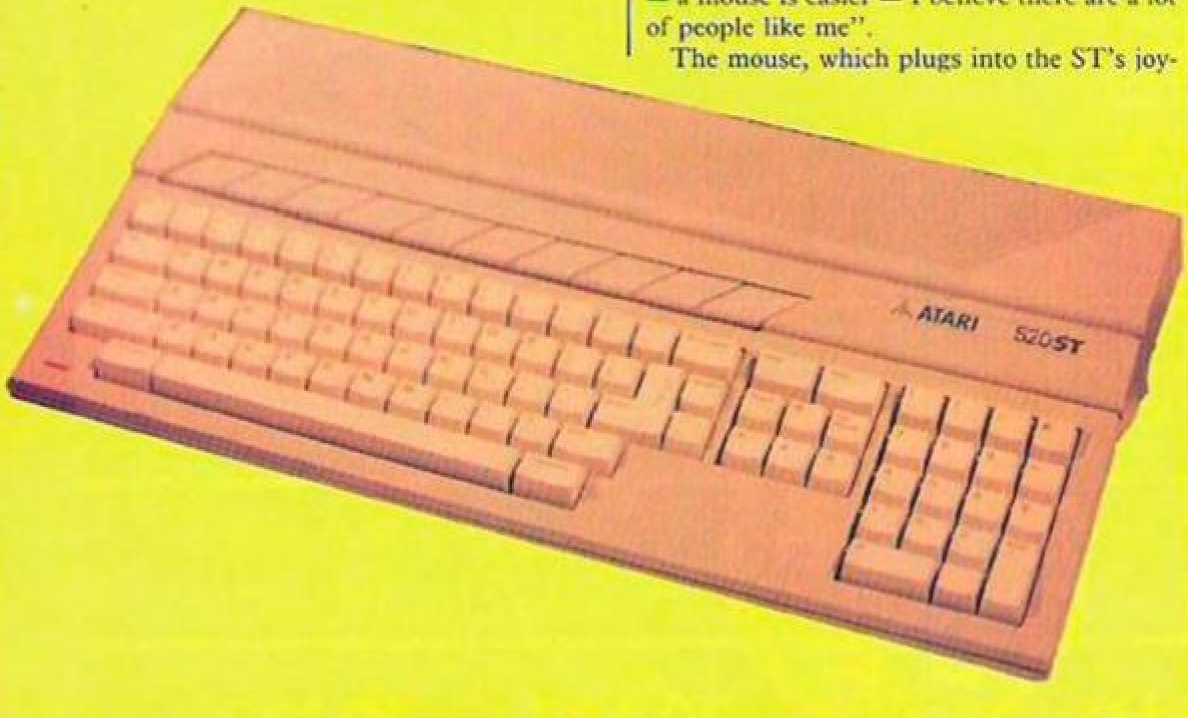
The screen on a Wimp machine is covered with "icons" — images representing the sort of objects you might find scattered across a desk, a calculator, files, even small pictures of floppy discs, hard discs, a working clock and a dustbin.

In front of you, on your desk, is a mouse which behaves like an inverted track ball with a couple of buttons on it. With its size and the "tail" — the lead which connects it to the micro — you can see how it gets its name. As you move the mouse around on your uncluttered real desk you control a pointer which runs around the simulated desk on the screen.

As you position it over an icon a touch of a button will call a file into operation or open up new possibilities. You can window sections of different files on the screen at the same time giving you the same flexibility you would have with various sheets of paper which you might want to compare on a real desk.

At best the Wimp approach makes using your micro quicker and easier as well as more fun but you still have to enter text through the keyboard and some of the icons are really just gimmicks. Using the calculator by positioning the mouse to put the pointer over each key in turn is infinitely slower and more inaccurate than using the numeric keypad of the ST.

Although the Macintosh is far more expensive, the ST does not seem to have any poorer implementation of Wimp. Indeed, whereas



CATCH THE MAC

the Mac only has a monochrome screen one of the new Atari slogans is "there's more to life than black and white" and so the icons are picked out in white on a green desk and red can be used as an overlay.

The STs, or Saints if you forget to type a capital T, have three graphics modes.

In high resolution, which is monochrome, and medium which gives you four colours. High is 640x400 pixels and medium 640x200 while in low resolution, 320x200, you can put 16 colours on screen at once.

The use of Gem on the new Ataris is not a trade-off for reduced Ram or processing power. Like the Macintosh, the Saints use the full Motorola 68000 16-bit chip rather than the 68008 with the sawn-off address bus used in the Sinclair QL. Even the bottom of the range 130 SL will come with 128K Ram and the Tramiel Operating System which includes Gem is all included in the massive 192K Rom. You will have a choice of Basic or Logo included with the machine or both for a small extra fee.

The number of interfaces built in suggests a commitment to the real world. You can plug in a 128K cartridge — on the prototype they used this for the constantly revised operating system — but Atari insists that by the time the machine goes into production there will be no need for a QL style dongle hanging out of the side. There is a floppy-disc interface with on-board controller, and also a port for the 15 megabyte hard disc which Tramiel hopes to sell for around £400.

He points out that hard discs could change the nature of your computer into a home data-bank — "You could store for instance the whole records of law for the last 200 years at home". The Saint has three-channel sound plus noise generator with separate frequency, volume and envelope controls but it also has a Midi interface which will allow you to plug it into most new synthesisers and keyboards.

The STs also have ports for Centronics printer, RS 232 modem, two joysticks — one set up for the mouse, TV, composite video, RGB and high-res monochrome monitor.

The 130 ST will be sold with its mouse for round £350 while the £500 512K 520 ST will be mainly sold in a package with disc drive, printer and monitor for around £1,000.

For the specification these prices seem almost unbelievably cheap but Tramiel insists that there is no sting in the tail, "I don't know why you're saying it's cheap, it will be a very profitable product — every computer I've sold I've made money".

Both Sinclair and Apple have found that software houses have taken a whole year to start producing programs for their 16-bit machines. Tramiel says that 25 houses are already working on software for the ST and Digital Research distributed the Gem system to American companies in November and British companies in February so at least the work has already started.

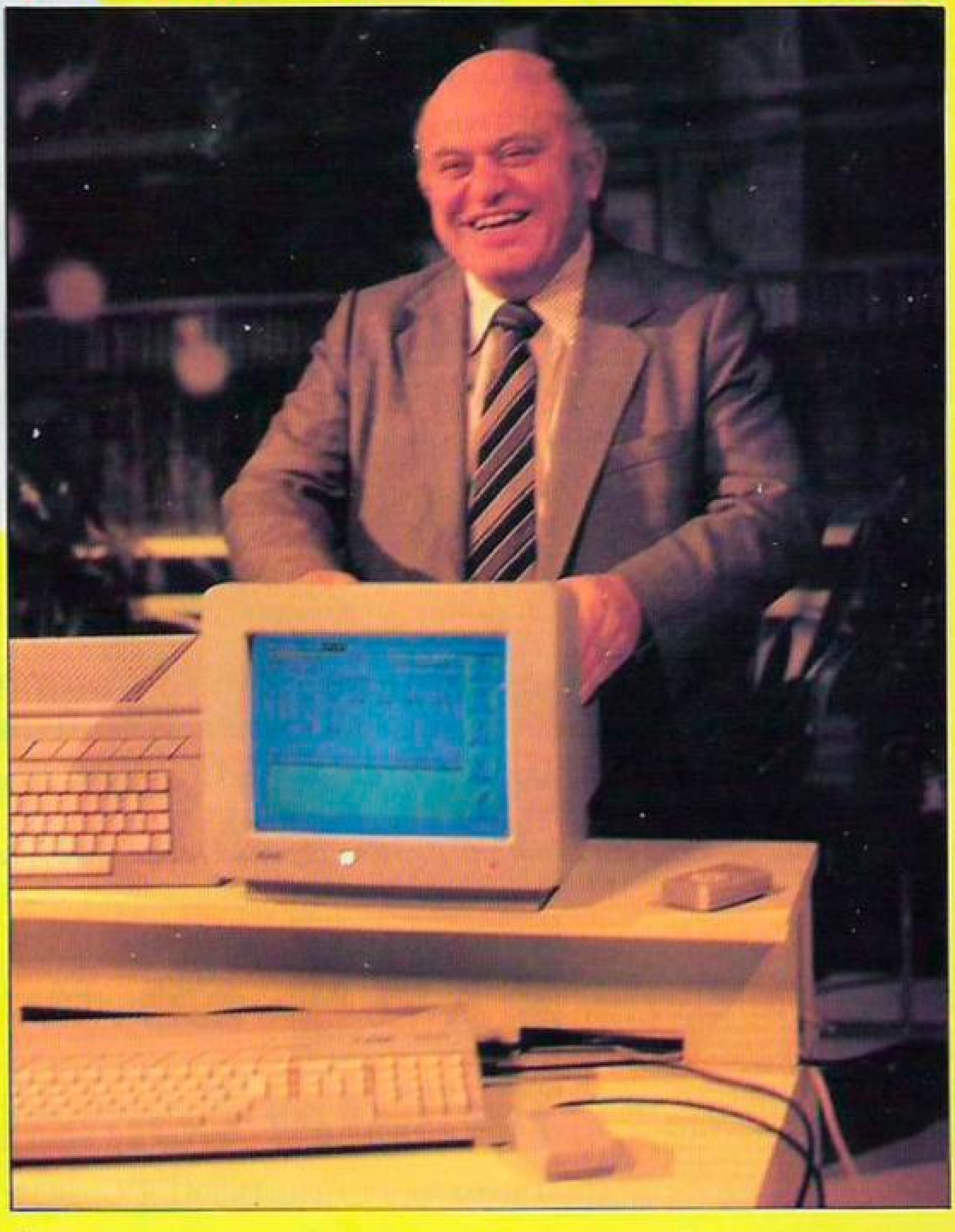
Atari's software supremo Siggy Hartman has also been over in Britain trying to organise British companies who want to write for the ST yet encouraging 6502 specialists to keep writing for Atari's repackaged eight-bit XE computers which are variations on the 800XL.

The Infinity package for the XL is due soon — word processor, spelling checker, database and spreadsheet, which should go down well with people who have just opted for the 800 XL plus disc drive offer. Atari will be encouraging everyone to move over to discs in 1985. Siggy says cassettes are "gonna be a thing of the past. The whole market is shifting into disc drives. Prices have to come down. Why should the consumer pay £200 for them? We're leading the way in this area, as we have done in others, by making sure we volume produce at prices the consumer can afford."

Specialist software for the 128K £200 130 XE, and the portable and music micro versions of the 65 XE (AKA 800 XL) is promised as soon as those machines are released later in the year.

If Atari can keep all of Tramiel's promises for this year, the opposition will have two choices — sit back and watch Jack clean up — he aims to be Number One within a year — or bring in radical new products themselves. Despite his original misgivings about mice and men Sinclair is now thought to have authorised modifications to turn the QL into a Wimp. Although Commodore is shouting loudest about the traditional but ultra-compatible CBM 128 "Shotgun" at the moment it is expected to release its new Amiga Lorraine-based 68000 micro at the Hanover Fair later this month although at a higher price than the Ataris.

Either way the cassette-based games only home computer may be on its last legs by the end of the year. We can expect to see more controversial attacks like Sir Clive's on Tramiel as the war hots up, but Jack refuses to reply in kind because "I'm afraid I might see him in a bar so I don't want to say anything bad". ■



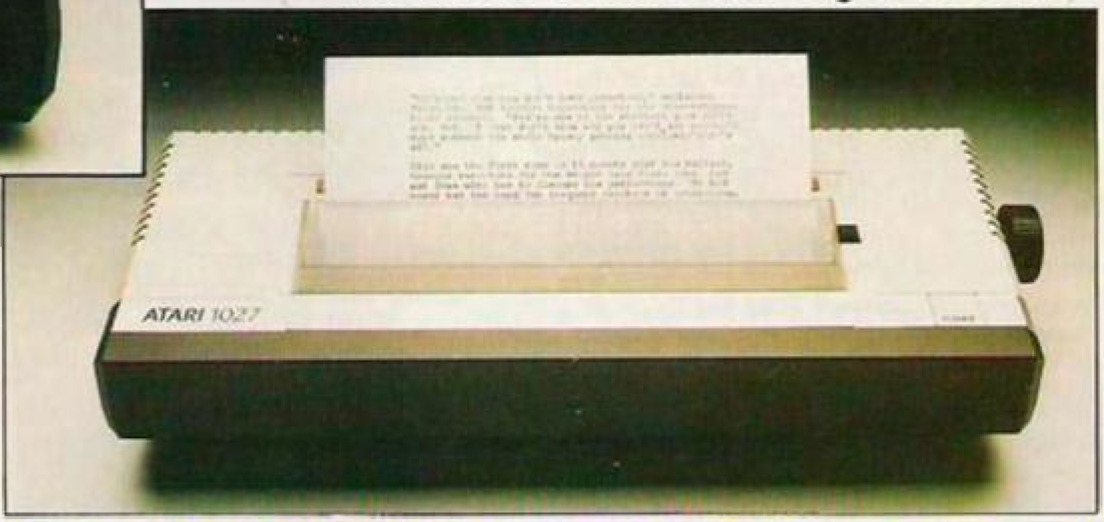
ATARI SYSTEM *competition*





Win an Atari 800XL 64K disc and printer system. Just compose an Atari-related limerick like this one.

Whenever Clive went to parties
He'd crawl on the floor eating Smarties
His friends understand
He's obsessed with Pac-Man
And turns blue at the sight of Ataris



RULES

- The winner of the competition will be the person who, in the view of the editor, comes up with the most apt and amusing limerick about Atari.
- The name of the winner will be printed in the July issue of *Your Computer*.
- All entries must arrive at the *Your Computer* offices by the last working day in April 1985.
- Each person may enter the competition only once.
- Entries to the competition cannot be acknowledged.
- No employees of Business Press International or their relatives may enter the competition.
- The decision of the editor is final.
- No correspondence on the result of the competition will be entered into.
- Business Press International assumes no responsibility or liability for any complaints arising from this competition.

YOUR COMPUTER COMPETITION

Don't forget to enclose this coupon, or a photocopy of it, when you send your limerick to *Your Computer*, Room L221, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS.

Name ____________________

Address ____________________

DEAR HEALTHSCREEN...

John Dawson highlights the potential health problems encountered by people who work with micros.

MANY OF YOU wrote to us about your health experiences with computers following the article 'Seriously: Computers damage your Health' in the January issue. The worries you outlined included eye strain and deteriorating eyesight, inflamed sinuses, an increased susceptibility to minor allergies, headaches, mild 'panic' attacks, loss of energy, arteriosclerosis (hardening and narrowing of the arteries) in the legs, increased consumption of cigarettes, and increased abortion rates in pregnant women.

That's quite a list of problems. If some of them are in fact connected to using a Visual Display Unit, it's a serious matter.

A Visual Display Unit (VDU) is a Cathode Ray Tube (CRT) mounted in a box with some associated electronic circuits. Most of the circuits in a modern VDU use the ordinary low voltages necessary for operating transistors. The deflection coils in a VDU intended for use with a microcomputer, for example, are current operated producing a magnetic deflection field, whereas a general purpose oscilloscope uses high voltages to deflect the electron beam electrostatically.

There is one source of very high voltage in any CRT device and this is the extra high tension (EHT) voltage that accelerates the electron beam from the cathode at the back of the tube towards the screen at the front.

VDUs are used in offices, laboratories, hospitals, schools, and at home. In each of these places there will be different surroundings, different lighting conditions, different temperature and humidity, and different seating arrangements. The motivation that goes with the use of a VDU also changes according to the surroundings and the purpose for which it is being used.

Some of the complaints against the visual display unit can be eliminated at once. I know of no rational basis for believing that arteriosclerosis is caused by exposure to some influence from a VDU. The person who wrote that letter said that the condition appeared shortly after he had acquired his computer. He also said that his cigarette consumption is adversely affected by using the computer: "Cigarettes just disappear when I am working"

Stop smoking

It is almost certain that smoking cigarettes for a number of years has caused his arteriosclerosis, although he is correct in saying that sitting on a hard chair which cuts into the back of his thighs will restrict the blood flow in his legs even further, possibly worsening the condition. I'm sorry but it would be wrong to attribute these health problems to the computer or the VDU.

Other correspondents wrote about soreness, redness and irritability around their eyes. When you concentrate on looking at a VDU your eyes are being asked to carry out an abnormal task. Normally you alter the tension on the lens in the eye at frequent intervals as you look around a room, look down at the papers you are working with, or gaze out of the window. This cycle of contraction and relaxation of the muscle surrounding the lens in the eye can be disrupted when you stare for a long period at a VDU screen.

The colour of the VDU screen worried one

Computers can damage your health! — Dr Chandra helps HAL 9000 in a scene from MGM/UA's 2010.



Eyes

If you find that you experience frequent headaches or disturbance of your vision which is associated with using a visual display unit you should consult a properly qualified ophthalmic optician. The optician may be able to reassure you that the problem is not connected with using the VDU. Alternatively, you may have a hidden visual problem which is being brought out by the particular circumstances, for example, of focussing at close distances for prolonged periods. The optician may be able to correct a defect in your vision or may suggest that you see a doctor.

Generally, you can help yourself by looking round the room so that you focus on objects that are at different distances, and by moving your eyes so that you are not simply gazing straight ahead down a narrow tunnel to the screen. Try to position the VDU so that you are not troubled by reflections or glare from pinpoint sources of light. Try to adjust the lighting in the room so that it's at a comfortable level and you aren't straining to read the documents that you are typing from.

Posture

Good office furniture, that is shaped in the right way, provides support for your back and places you at a comfortable height so that you can type or operate the computer without unnecessary fatigue in your fingers and allows room for your thighs under the desk. Home computer arrangements will often fall short of a well designed office; however, there are some things you can do to make things better.

Make sure that you can adjust the position of the VDU in relation to the keyboard. Ideally you should be able to move the VDU up, down, side to side, and by rotating the screen. If you are copy typing for significant periods try to keep the papers you are typing from at much the same height and angle as the VDU screen. Place them as close as possible to the side of the VDU so that you don't have to twist your head to look from the screen to the papers and back again.

Taking a break

Sitting still in a chair for hours on end never did anyone much good. Try to take a break at least once an hour for at least a couple of minutes. Two minutes every half an hour is probably a sensible ratio. Get up and walk round to get the muscle pump in your legs working again; thrombosis in your leg veins can be caused by the edge of a chair pressing on the back of your calves or thighs for long periods — as in a deck chair in summer, it isn't just work that's a health hazard!

Smoking

Smoking will knock your ability to concentrate and will make some of the risks outlined above much greater.

correspondent in particular. She could work with a black and white screen quite satisfactorily but green screens gave her headaches and she blames these for a deterioration in her eyesight. As far as the colour is concerned it's possible that the green screens may have flickered more than the others.

The face of a cathode ray tube, the part on which the picture appears, is coated inside the tube with a chemical mixture with fluoresces when it is struck by electrons. The coating in different tubes is modified especially for each different purpose; a radar tube, for example, is designed so that the chemical coating (the phosphor) will continue to glow for a long time (about as long as it will take for the trace to sweep a complete circle) so that an image of a ship or the coastline will be continuously visible. An oscilloscope that will display a pulse that is a few nanoseconds wide has an entirely different type of phosphor coating.

If a high persistence phosphor is used in a VDU the display will seem stable and pleasant to use for static text or displays but you will see smudging when text scrolls or a fast moving shape bursts across the screen. Amber displays often have a slightly longer image decay time and are often recommended for word processing and other serious applications. Green displays vary greatly and you should check the manufacturer's specifications if it is important to you. Amateur colour VDUs are all built using CRTs that will handle colour television pictures. The response time has to be short enough to cope with a picture that may change totally in one twenty-fifth of a second and you may be able to detect some flicker, especially if you look at the VDU out of the corner of your eye.

Allergic reactions

If you look at other people you can see that in ordinary circumstances we blink several times each minute. A person's blink rate may be an indicator of their state of tension but it is also altered by concentration on tasks requiring fine visual discrimination. Pat Hawker has described how early radar operators in the Second World War suffered from styes. These minor infections may have resulted from the conjunctiva, the layer of tissue covering the outer surface of the eye, suffering some damage as a consequence of changes in the normal blink rate.

There seems little doubt now that using a VDU can unmask hidden opthalmic conditions. If you have trouble moving your eyes so that they both point at a near object (convergence), or difficulty in producing a stereoscopic image, this may show up when you use a VDU. The outward signs that you may expect from this group of complaints could include headaches, difficulty in focussing on distant objects after using a VDU for some time and persistent soreness round the eyes. The symptoms that one reader described do suggest that she should have her eyes examined.

However, while the machinery itself produces an artificial situation that may expose pre-existing problems, it will not damage a normally sighted person.

The circumstances in which a VDU is used are also very important. The lighting in many British offices is badly designed and may be frankly inadequate. Natural lighting from windows will be insufficient at times in the winter and may be too bright, or cause too much glare in the summer months. Artificial light should be adjusted to the nature of the work you have to do. Some tasks require much higher light levels, to help you to pick out fine detail, for example, than others.

A manual on the use of Visual Display Terminals by the Illuminating Engineering Society recommends that no more than 500 lux — a measure of the brightness of the lighting at a particular place like the surface of a desk — and no less than 300 lux illuminance should be provided as measured on the surface on which the VDU is standing (Designing Systems for People, L Damodoran, National Computing Centre, 1980). This is rather less than a well-lit typing centre (750 lux) and more than is necessary in a filing room (300 lux). Avoiding glare and reflections from the screen of the VDU are very important if eyestrain is to be avoided.

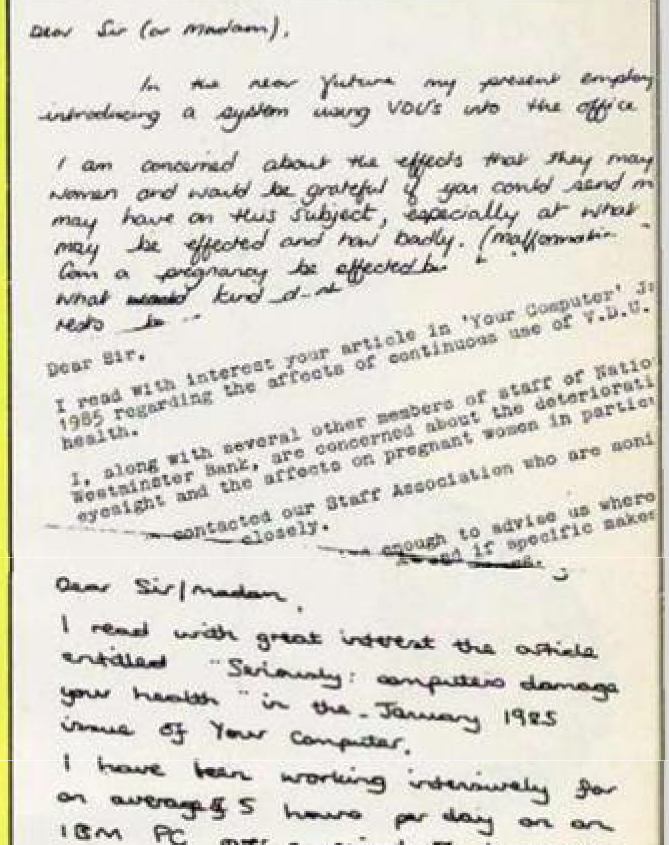
The minor allergies that one correspondent reported may be caused by a VDU. This is not proven yet but there is a theory that the EHT voltage that drives the CRT causes a diminishing static potential in front of the face of the tube. A person sitting a couple of feet from the VDU may accumulate dust on their skin as a result of the static discharge from the tube to their face. The dust can cause irritation, and this, coupled with the abrasion caused by rubbing the face, can cause an allergic reaction. The static build up on the CRT will depend to some extent on the humidity in the office or workroom.

The X-rays produced by VDUs are a fertile source of speculation and misinformation. If you accelerate electrons from a cathode and allow them to bombard a metallic anode X-rays will be produced. Roentgen discovered X-rays almost by chance in the latter part of the last century. As X-ray technology developed, special tubes were made that produced a concentration of X-rays in one direction. Ordinary monochrome CRTs produce no significant X-ray hazard that I know of. Some studies have shown a potentially significant emission of X-rays from colour cathode ray tubes which has been ascribed to the higher accelerating voltages (EHT).

Broken marriages

If you work with VDUs and you are concerned about X-ray emission probably the best people to advise you are the local officers of a trade union such as APEX, NALGO or ASTMS. These unions have access to research on the subject and have model policies to regulate exposure both from the point of view of women who may be pregnant and, more generally, users who may experience eye trouble. If you are pregnant, or if you are planning to become pregnant, then it makes sense to reduce the risks to the baby as far as possible. To do this sensibly you must have some idea of the size of the risk and smoking, for example, is an enormously greater health hazard to an unborn child than operating a computer. There is no comparison between the two dangers — so keep it in perspective.

One person who wrote to us mentioned, half jokingly, that when his wife divorced him it would be because of the time he spent "working" on his computer. At least he was honest. it is remarkable how compulsive programming or operating a computer can become. A study is under way at Loughborough University to examine the scale and



nature of the problem. I have seen people who have become withdrawn and antisocial as a result of working on a program. If they have a regular job their work suffers, and personal relationships can be corroded beyond repair. The jokes about being up all night, not noticing the time and so on, are funny for some people and sadly accurate for others.

From isolated reports it seems that marriages really have been broken by one partner, usually the man, having access to a computer, personal or mainframe, and not being able to handle the compulsion to "just try it once more and it'll come out right".

This article has covered quite a few of the worrying or downright negative aspects of computing. It's worth remembering that small computers are being used positively to help people all over the country who are physically handicapped. Word processors with special input devices, programs to assist the development of better coordination between a child's eyes and hands, computer-based aids to help deaf children to learn to speak are among many use for computers — general purpose programmable machines — to improve health. Far more could be done by sharing information that would increase the benefits from microcomputing while simultaneously avoiding as many of the risks as possible. ■

Instead of computers catch
technology now has to cat



The way we see it, technology has suddenly got quite a race on its hands.

There's no other home computer in the world that's so expandable or so updatable as the new Enterprise 64.

And if you're wondering quite how we've managed that, kindly take a closer look at the outputs on our remarkable new machine.

You'll notice that amongst all the usual sockets and terminals, we've gone and incorporated a special 66-way expansion port.

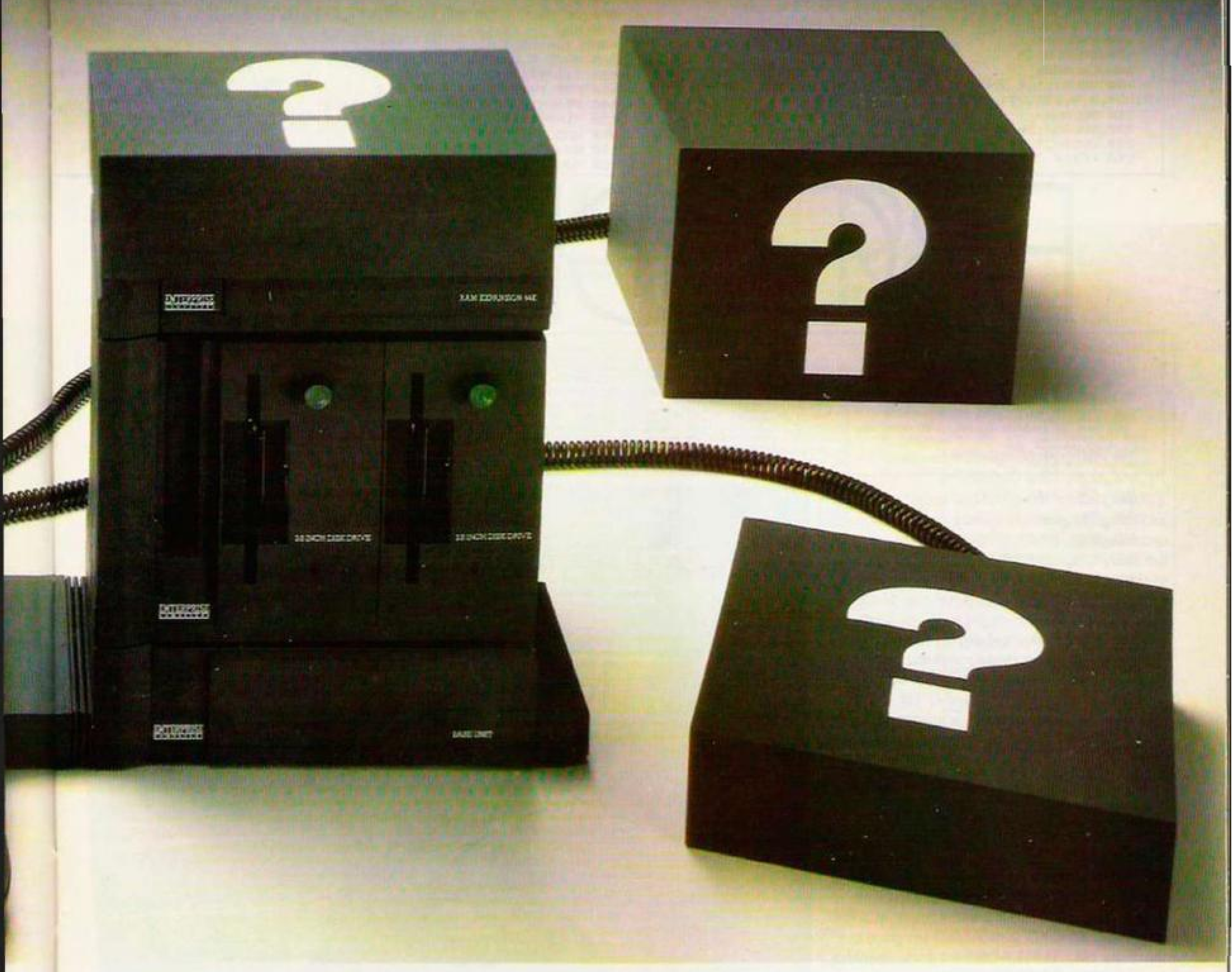
This will accept a whole range of new peripherals that are in the pipeline. Including those that are a mere twinkle in the eyes of our hardware designers.

We thought this expandability principle was such a good idea, we applied it to the Enterprise's memory, too.

Even in its most basic 64K form, this puts more user RAM at your disposal than almost any other competitor.

But plug in our special Rampacks to the base unit, and you can progressively increase that figure to a truly extraordinary 3,900K.

c hing up with technology,
a tch up with a computer.



Not that that's the only challenge we present to today's ambitious programmers.

With a screen resolution of up to 672 x 512 pixels, 256 colours and a high speed video processor, the Enterprise will outgun all but the highest quality TV monitors.

And the sophisticated sound chip generates no fewer than 4 voices across 8 octaves in full stereo.

Combine the two and you can create effects that leave today's games looking like pub video tennis of the mid-seventies.

For anyone with literary aspirations, the Enterprise also comes complete with an integrated word processor.

Whilst the really serious user will be delighted to discover analogue RGB and TV outputs, as well as parallel, RS423 serial and network ports.

Both Cobol and 'C' will be available with CP/M running, and you can even use Lisp, Forth and Z80 assembly language on cartridge without encroaching on user RAM.

The new Enterprise 64.

It hasn't just overtaken technology. It's left every other home computer straggling in the distance.

ENTERPRISE
C O M P U T E R S

WITH OBSOLESCENCE BUILT-OUT

```
Listing 2.
   10 REM ::::::::::::::::::::::::::::::::::::
   20 REM :::::::    GRAND PRIX  v1   ::::::::
   30 REM ::: (c) Michael Barter 1984 ::::::
   40 REM ::::::::::::::::::::::::::::::::::::
   50 ENVELOPE1,3,0,0,0,0,0,0,126,-15,0,-5,126,100
   60 *FX200,1
  100 !&83=!RND(!RND)
  110 IF !&83=0 THEN 100
  120 MODE2
  121 *FX 9,2
  122 *FX 10,2
  130 VDU23,1,0;0;0;0;
  140 PROCvdu
  150 PROCsky
  160 PROCroad
  170 PROCcol
  180 COLOUR134
  190 COLOUR4
  200 VDU31,1,0,230,231,232,9,9,224,225,226,9,9,227,22
8,229,9,9,233,234,235,9
  210 COLOUR7
  220 COLOUR128
  230 VDU28,3,31,16,29
  240 *FX19
  250 *FX14,5
  260 *L."Chars"
  270 */"Prix"
  280 END
  285 :
  290 DEFPROCsky
  300 GCOL0,6
  310 MOVE0,600:MOVE0,1024:PLOT85,1280,600
  320 PLOT85,1280,1024
  330 FOR CLOUDS=1 TO 3+RND(4)
  340 Y%=700+RND(200)
  350 PROCCLOUD(200+RND(870),Y%,100+RND(100),30+RND(20
7,6)
  360 NEXT
  370 ENDPROC
  380 DEFPROCCLOUD(X%,Y%,SX%,SY%,C1%,C2%)
  390 VDU 29,X%;Y%;
  400 L%=6+RND(8)
  410 MOVE 0,0:MOVE SX%+SX%/L%,0
  420 X1%=SX%+SX%/10:Y1%=0
  430 FOR I=0 TO 6.4 STEP 0.2
  440 X%=SX%*COS(I)+SX%/L%*COS(I*L%)
  450 Y%=SY%*SIN(I)+SY%/L%*SIN(I*L%)
  460 GCOL 0,C1%
  470 MOVE 32,12:PLOT 85,X%,Y%
  480 MOVE X1%,Y1%
```

BBC GRAND

```
Listing 1.
   10REM :::::::::::::::::::::::::::::::::::::::
   20REM ::::::::::::Grand Prix ::::::::::::::::
   30REM ::::: (c) Michael Barter 1984  ::::::
   40REM :::::::::::::::::::::::::::::::::::::::
   50MODE7:IF INKEY-256<>0 AND INKEY-256<>-1 THEN PRIN
T"This program is only for a BBC Microcomputer sorry e
lectron users":END
   60IF INKEY-256 A%=&EA:X%=0:Y%=&FF:IF(USR(&FFF4) AND&
FF00 THEN PRINT"Tough luck " This program wont work wi
th a second processor":END
   70IF?&FE64=?&FE64 PRINT"IC 69 missing / faulty - Pl
ease go out and buy one":END
   80A%=129:X%=0:Y%=255:IF(USR&FFF4 AND&FF00)=0 THEN PR
INT"Sorry but this program only works with OS 1.2":END
   90PROCheader("G r a n d  P r i x")
  100PROCcentre(CHR$133+"(c) Michael Barter 1984",3)
  110 PRINT
  120PROCdouble(CHR$134+"Controls")
  130PROCjustify("Z - Left",8)
  140PROCjustify("X - Right",9)
  150PROCjustify(": - Speed up",10)
  160PROCjustify("/ - Speed Down",11)
  170PROCjustify("Q - Sound off",12)
  180PROCjustify("S - Sound on",13)
  190PROCjustify("P - Pause",14)
  200PROCjustify("R - Resume",15)
  210PRINTTAB(0,20)"      You must get a Bonus of over
70 to qualify for the next stage. If you stay still to
 long the other cars will crash into you."
  220VDU28,10,19,28,17
  230PAGE=&2500
  240CHAIN"Grand"
  250DEFPROCcentre(A$,vpos)PRINTTAB((20-LEN(A$)DIV2),vpo
s);A$;:ENDPROC
  260DEFPROCheader(A$)VDU30:FORI=0TO1:PRINTCHR$141;CHR
$157;CHR$131;CHR$130;:PROCcentre(A$,VPOS):NEXT:ENDPROC
  270DEFPROCjustify(A$,vpos)PRINTTAB(15,vpos);A$:ENDPR
OC
  280DEFPROCdouble(A$)FORI=0TO1:PROCcentre(CHR$141+A$,
VPOS):NEXT:ENDPROC
```

Michael Barter speeds up to the starting grid with a winner for the BBC.



YOUR CHANCE TO join the kings of speed, jockeying for position as they push their sleek technological thoroughbreds to the utmost limits. No, it's not the South Circular at 8.30 in the morning, but a Grand Prix race simulation for the BBC Model B, or the Model A plus 32K. But restrain the urge to strap on your leather goggles and gloves for a bit; keying a program in is already awkward enough.

The number in line 10270 is the number of cars you have to pass to qualify for the next stage *10. At the moment this number is set to 7 which means you have to pass 70 cars. This number should be lowered to make the game easier. There are five different stages to the game which include countryside, snow, desert, night and riverside scenes.

Type in listing 1 and save it on tape then set page to &2500 and type in listing 2 and save it on the same tape as listing 1. To type in listings 3 and 4 page will also have to be set to &2500. Type in listing 3 and save it on a different tape to the first two. This program contains the data for the machine-code graphics and could corrupt itself when run.

Run listing 3 and if all is well a prompt will appear. You can now press space to save the graphics data on tape along with the other two listings. This process will have to be repeated to type listing 4. The programs should be saved with the following filenames Title, Grand, Chars, Prix.

Typing in the program, especially listing 4, will be a long process so for anybody who doesn't have the time I will send it to them on a high-quality C-15 cassette if they send me a cheque for £3.50 made payable to M. Barter at 105 Lewis Street, Crumlin, Gwent NP1 5EF.

```
490 GCOL 0,C2%:DRAW X%,Y%
500 X1%=X%:Y1%=Y%
510 NEXT
520 VDU 29,0;0;
530 ENDPROC
540 DEFPROCroad
550 GCOL0,2
560 MOVE0,0:MOVE0,600:PLOT85,1280,0
570 MOVE1280,0:MOVE1280,600:PLOT85,0,600
580 X%=10:COL=1
590 FOR L%=1TO 20
600 GCOL0,11+COL
610 MOVE 0,605-X%:DRAW1279,605-X%
620 X%=X%+X%/2
630 COL=COL MOD3+1
640 NEXT L%
650 GCOL0,0:MOVE0,0:MOVE580,600:PLOT85,1280,0:PLOT85
,700,600
660 T%=1:COL=1
670 FORN=0 TO 600
680 GCOL0,8+COL
690 PLOT77,650,N
700 T%=T%+1:IF T%=50 THEN COL=COL MOD3+1:T%=0
710 NEXT
720 GCOL0,0
730 MOVE100,0:MOVE600,600:PLOT85,1180,0
740 PLOT85,680,600
750 ENDPROC
760 DEFPROCvdu
770 VDU23,224,247,132,132,135,244,20,20,244
780 VDU23,225,187,162,162,187,34,34,34,59
790 VDU23,226,184,36,36,164,36,36,36,184
800 VDU23,227,250,170,34,34,34,34,34,34
810 VDU23,228,139,218,170,139,138,138,138,139
820 VDU23,229,128,0,0,128,0,0,0,128
830 VDU23,230,247,132,132,132,244,20,20,247
840 VDU23,231,123,74,74,75,74,74,74,122
850 VDU23,232,156,80,80,156,80,80,80,92
860 VDU23,233,231,148,148,228,148,148,148,231
870 VDU23,234,165,165,165,181,173,165,165,164
880 VDU23,235,47,40,40,47,33,33,33,207
890 ENDPROC
900 DEFPROCcol
910 VDU19,9,7;0;
920 VDU19,10,0;0;
930 VDU19,11,0;0;
940 VDU19,12,2;0;
950 VDU19,13,2;0;
960 VDU19,14,3;0;
970 ENDPROC
```

PRIX RACER

Listing 3.

```
10REM ***    Grand Prix PART 3    **
20REM ** (c) Michael Barter 1984 **
30FOR I%=0 TO 1638
40READchar$
50I%?&900=EVAL("&"+char$)
60NEXT
70PRINT" PRESS SPACE TO SAVE":REPEAT
  UNTIL GET$=" "
80*SAVE"Chars"0900+0667 0900
90GOTO 80
100DATA00,00,15,3F,00,00,00,00,00,3F,2A,00,00
110DATA00,00,00,2A,00,00,00,00,00,15,00,00,00
120DATA00,00,15,15,33,33,33,33,33,2A,2A,00,00
130DATA00,00,00,00,33,33,33,33,33,00,00,00,00
140DATA00,00,15,3F,33,33,33,33,33,3F,2A,00,00
150DATA00,00,00,33,33,33,33,33,33,33,00,00,00
160DATA00,01,03,03,03,03,03,03,03,03,03,02,00
170DATA00,00,03,03,03,03,03,03,03,03,00,00
180DATA00,00,33,23,33,33,33,33,33,13,33,00,00
190DATA00,3F,3F,23,31,30,30,30,30,13,3F,3F,00
200DATA00,6A,6A,23,3F,3F,3F,3F,3F,13,95,95,00
210DATA00,55,55,3C,3C,3C,3C,3C,3C,3C,AA,AA,00
220DATA00,00,00,3C,3C,3C,3C,3C,3C,3C,00,00,00
230DATA00,00,00,22,00,00,00,00,00,11,00,00,00
240DATA00,00,00,36,3C,3C,3C,3C,3C,39,00,00,00
250DATA00,00,00,22,00,00,00,00,00,11,00,00,00
260DATA00,00,00,11,33,33,33,33,33,22,00,00,00
270DATA00,FF,FF,00,00,00,00,00,00,00,FF,FF,00
280DATA00,C0,C0,00,00,00,00,00,00,00,C0,C0,00
290 REM" Enemy Car **
300DATA00,00,00,00,00,00,00,00,00,00,00
310DATA00,00,15,3F,00,00,00,3F,2A,00,00
320DATA00,00,00,2A,00,00,00,15,00,00,00
330DATA00,00,15,15,30,30,30,2A,2A,00,00
340DATA00,00,00,00,30,30,30,00,00,00,00
350DATA00,00,15,3F,30,30,30,3F,2A,00,00
360DATA00,00,00,30,30,30,30,30,00,00,00
370DATA00,01,03,03,03,03,03,03,03,02,00
380DATA00,00,03,03,03,03,03,03,03,00,00
390DATA00,00,30,21,30,30,30,12,30,00,00
400DATA00,3F,3F,21,3F,3F,3F,12,3F,3F,00
410DATA00,6A,6A,2A,3F,3F,3F,12,95,95,00
420DATA00,55,55,3C,3C,3C,3C,3C,AA,AA,00
430DATA00,00,00,3C,3C,3C,3C,3C,00,00,00
440DATA00,00,00,20,00,00,10,00,00,00,00
450DATA00,00,00,34,3C,3C,3C,38,00,00,00
460DATA00,00,00,20,00,00,00,10,00,00,00
470DATA00,00,00,10,30,30,30,20,00,00,00
480DATA00,FF,FF,00,00,00,00,00,FF,FF,00
490DATA00,C0,C0,00,00,00,00,00,C0,C0,00
500REM" Enemy Car ***
510DATA00,00,00,00,00,00,00,00,00,00,00,00
520DATA00,00,15,3F,00,00,00,00,3F,2A,00,00
530DATA00,00,00,2A,00,00,00,00,15,00,00,00
540DATA00,00,15,15,30,30,30,30,2A,2A,00,00
550DATA00,00,00,00,30,30,30,30,00,00,00,00
560DATA00,00,15,3F,30,30,30,30,3F,2A,00,00
570DATA00,00,00,30,30,30,30,30,30,00,00,00
580DATA00,01,03,03,03,03,03,03,03,03,02,00
590DATA00,00,03,03,03,03,03,03,03,03,00,00
600DATA00,00,30,21,30,30,30,30,12,30,00,00
610DATA00,3F,3F,21,3F,3F,3F,3F,12,3F,3F,00
620DATA00,6A,6A,21,3F,3F,3F,3F,12,95,95,00
630DATA00,55,55,3C,3C,3C,3C,3C,3C,AA,AA,00
640DATA00,00,00,3C,3C,3C,3C,3C,3C,00,00,00
650DATA00,00,00,20,00,00,00,00,10,00,00,00
660DATA00,00,00,34,3C,3C,3C,3C,38,00,00,00
670DATA00,00,00,20,00,00,00,00,10,00,00,00
680DATA00,00,00,10,30,30,30,30,20,00,00,00
690DATA00,FF,FF,00,00,00,00,00,00,FF,FF,00
700DATA00,C0,C0,00,00,00,00,00,00,C0,C0,00
710REM" Enemy Car ****
720DATA00,00,00,00,00,00,00,00,00,00,00,00,00
730DATA00,00,15,3F,00,00,00,00,00,3F,2A,00,00
740DATA00,00,00,2A,00,00,00,00,00,15,00,00,00
750DATA00,00,15,15,30,30,30,30,30,2A,2A,00,00
760DATA00,00,00,00,30,30,30,30,30,00,00,00,00
770DATA00,00,15,3F,30,30,30,30,30,3F,2A,00,00
780DATA00,00,00,30,30,30,30,30,30,30,00,00,00
790DATA00,01,03,03,03,03,03,03,03,03,03,02,00
800DATA00,00,03,03,03,03,03,03,03,03,03,00,00
810DATA00,00,30,21,30,30,30,30,30,12,30,00,00
820DATA00,3F,3F,21,3F,3F,3F,3F,3F,12,3F,3F,00
830DATA00,6A,6A,21,3F,3F,3F,3F,3F,12,95,95,00
840DATA00,55,55,3C,3C,3C,3C,3C,3C,3C,AA,AA,00
850DATA00,00,00,3C,3C,3C,3C,3C,3C,3C,00,00,00
860DATA00,00,00,20,00,00,00,00,00,10,00,00,00
870DATA00,00,00,34,3C,3C,3C,3C,3C,38,00,00,00
880DATA00,00,00,20,00,00,00,00,00,10,00,00,00
890DATA00,00,00,10,30,30,30,30,30,20,00,00,00
900DATA00,FF,FF,00,00,00,00,00,00,00,FF,FF,00
910DATA00,C0,C0,00,00,00,00,00,00,00,C0,C0,00
920REM" Enemy Car *****
930DATA00,00,00,00,00,00,00,00,00,00,00,00,00,00
940DATA00,00,15,3F,00,00,00,00,00,00,3F,2A,00,00
950DATA00,00,00,2A,00,00,00,00,00,00,15,00,00,00
960DATA00,00,15,15,30,30,30,30,30,30,2A,2A,00,00
970DATA00,00,00,00,30,30,30,30,30,30,00,00,00,00
980DATA00,00,15,3F,30,30,30,30,30,30,3F,2A,00,00
990DATA00,00,00,30,30,30,30,30,30,30,30,00,00,00
1000DATA00,01,03,03,03,03,03,03,03,03,03,03,02,00
1010DATA00,00,03,03,03,03,03,03,03,03,03,03,00,00
1020DATA00,00,30,21,30,30,30,30,30,30,12,30,00,00
1030DATA00,3F,3F,21,3F,3F,3F,3F,3F,3F,12,3F,3F,00
1040DATA00,6A,6A,21,3F,3F,3F,3F,3F,3F,12,95,95,00
1050DATA00,55,55,3C,3C,3C,3C,3C,3C,3C,3C,AA,AA,00
1060DATA00,00,00,3C,3C,3C,3C,3C,3C,3C,3C,00,00,00
1070DATA00,00,00,20,00,00,00,00,00,00,10,00,00,00
1080DATA00,00,00,34,3C,3C,3C,3C,3C,3C,38,00,00,00
1090DATA00,00,00,20,00,00,00,00,00,00,10,00,00,00
1100DATA00,00,00,10,30,30,30,30,30,30,20,00,00,00
1110DATA00,FF,FF,00,00,00,00,00,00,00,00,FF,FF,00
1120DATA00,C0,C0,00,00,00,00,00,00,00,00,C0,C0,00
1130REM" Explosion!
1140DATA00,00,05,0F,0F,0A,00,00
1150DATA00,01,00,03,03,07,0A,00
1160DATA01,07,03,07,1E,0B,0B,00
1170DATA05,0B,03,07,03,03,0F,00
1180DATA01,03,0B,23,03,03,07,02
1190DATA05,0B,07,0B,13,03,03,0A
1200DATA07,0B,0B,0B,03,13,03,0B
1210DATA03,07,13,07,0B,33,07,0B
1220DATA01,0F,16,1B,13,03,0F,00
1230DATA03,0F,03,03,13,03,0B,00
1240DATA0B,03,07,27,0B,0B,0B,00
1250DATA07,0B,03,03,03,07,02,00
1260DATA00,0B,03,07,0F,0F,0B,00
1270DATA00,07,0B,07,03,03,03,00
1280DATA01,01,1B,03,0B,02,02,00
1290DATA00,01,0F,0F,03,02,03,00
1300DATA00,03,03,07,00,00,01,00
1310DATA00,00,01,01,00,00,00,00
1320DATA00,01,01,00,00,00,00,00
1330REM"Numbers 0 to 9
1340DATA03,03,00
1350DATA02,01,00
1360DATA02,01,00
1370DATA02,01,00
1380DATA02,01,00
1390DATA02,01,00
1400DATA02,01,00
1410DATA03,03,00
1420:
1430DATA01,02,00
1440DATA01,02,00
1450DATA01,02,00
1460DATA01,02,00
1470DATA01,02,00
1480DATA01,02,00
1490DATA01,02,00
1500DATA01,02,00
1510:
1520DATA03,02,00
1530DATA00,01,00
1540DATA00,01,00
1550DATA00,01,00
1560DATA00,02,00
1570DATA01,00,00
1580DATA02,00,00
1590DATA03,03,00
1600:
1610DATA03,03,00
1620DATA00,01,00
1630DATA00,01,00
1640DATA03,03,00
1650DATA00,01,00
1660DATA00,01,00
1670DATA00,01,00
1680DATA03,03,00
1690:
1700DATA02,00,00
1710DATA02,00,00
1720DATA02,02,00
1730DATA02,02,00
1740DATA03,03,00
1750DATA02,02,00
1760DATA02,02,00
1770DATA00,00,00
1780:
1790DATA03,03,00
```

(listing 3 continued on next page)

(listing 3 continued from previous page)

```
1800DATA02,00,00
1810DATA02,00,00
1820DATA03,03,00
1830DATA00,01,00
1840DATA00,01,00
1850DATA00,01,00
1860DATA03,03,00
1870:
1880DATA02,00,00
1890DATA02,00,00
1900DATA02,00,00
1910DATA03,03,00
1920DATA02,01,00
1930DATA02,01,00
1940DATA02,01,00
1950DATA03,03,00
1960:
1970DATA03,03,00
1980DATA00,01,00
1990DATA00,01,00
2000DATA00,02,00
2010DATA00,02,00
2020DATA01,00,00
2030DATA01,00,00
2040DATA02,00,00
2050:
2060DATA03,03,00
2070DATA02,01,00
2080DATA02,01,00
2090DATA03,03,00
2100DATA03,03,00
2110DATA02,01,00
2120DATA02,01,00
2130DATA03,03,00
2140:
2150DATA03,03,00
2160DATA02,01,00
2170DATA02,01,00
2180DATA03,03,00
2190DATA00,01,00
2200DATA00,01,00
2210DATA00,01,00
2220DATA00,01,00
2230:
```

Listing 4.

```
10REM ********************************
20REM *                              *
30REM *          GRAND PRIX          *
40REM *   By Micheal John Barter     *
50REM *      Edited by Bryant.       *
60REM *                              *
70REM ********************************
80PROCas:PRINT"Press <SPACE> to save":REPEATUNTILGET=32:*SAVE"Prix"1020 2000 105A
120END
130DEFPROCas:timestore=&90
[illegible]
13360DATA11,00,00,00,00,00,00
13370DATA10,00,01,00,03,00,01,00
13380DATA00,00,01,00,06,00,02,00
```

HAVE YOU EVER wanted to mindlessly massacre millions of miserable morons, shoot or severely savage stupid spaceships or even angrily anihilate all available aliens? If your answer is yes to any of these then this game is for you. In this game you may fire away at any alien in sight and they won't even shoot back. What is more once you wipe out one batch you get another batch to take your anger out on — violent isn't it?

As the aliens don't shoot back this may sound like a simple game but unfortunately there is a certain great lumbering idiot who may shoot you — yourself. This is because not only can you shoot aliens with your beasty blasting bases, you can also blast holes in your tracks reducing your mobility or even worse, you can zap your own laser base.

This is all due to the screen layout of your bases. They are mounted on tracks on each of the four sides of the screen and controls work on pairs of bases. If you move your top base you also move your bottom base and if you fire you fire from all four bases. However if you fire and the bullet hits the opposite track, a hole forms and your laser base on that side can not move past it. If you shoot the hole again it fills it in.

The object of the game is to blast all the aliens into their component quarks, there are eight screens — freaky flying saucers, docile dunbells, artful asteroids, segmented centipedes, turning tops, bouncing balls, hysterial helicopters and slithering snakes. All these move in smooth pixel graphics and are animated.

On each screen there are eight aliens at a time. However, when you have killed, maimed or destroyed those eight, another eight appear to allow you to continue your vicious work. Each alien is worth a number of points corresponding to the level you are on. This number is also deducted from the scored titled "needed". This must be reduced to zero before you are allowed on to the next level and on each level the amount of points needed increases.

If you do not get enough points you restart the same round. You can only go on to the next level when you have lasted for 60 seconds on that screen — there is a bar chart under the screen showing the time. The game ends when all four of your bases are destroyed. You get a new set of four bases every four rounds you survive.

To type in the game type in listing 1 and save it with

SAVE "CROSSFIRE"LINE 1

Next type in listing 2 and run it. This pokes in half of the machine code. If the program detects an error it will tell what line it was in. However, it cannot detect certain errors and so it is best to keep a back up copy of it in case the code is wrong. If the program gives a "No errors" message, save the code with

SAVE "CODE1"CODE 30200,2480

Repeat this for listing 3 but save the code with

SAVE"CODE2"CODE 30300,2500

Now clear out the computer with

RANDOMIZE USR 0

and load both the codes with

CLEAR 27899:LOAD"CODE1"CODE 27900:.LOAD"CODE2"CODE 30380

Then save the whole code directly after the Crossfire program with

SAVE "CODE"CODE 27900,4867

The program is now ready to play.

SPECTRUM

CROSS FIRE

Listing 1.

```
   5 DEF FN x$(a)="WXYZ[\]↑_£"(I
NT (a/16)+1)+"WXYZ[\]↑_£"(a-16*I
NT (a/16)+1)
  10 BORDER 0: PAPER 0: INK 7: C
LEAR 27899: PRINT AT 7,5; BRIGHT
 1;"PLEASE WAIT,CROSSFIRE";AT 9,
12;"LOADING"
  15 DIM a$(6,3): DIM b$(6,6): F
OR a=1 TO 6: LET a$(a)="aaa": LE
T b$(a)="WWWWW"+CHR$ (94-a): NEX
T a
  20 LOAD "CODE"CODE 27900
  30 INK 7: POKE 23606,31: POKE
23607,124: CLS : PRINT INK 6;"CE
"'"DF";AT 0,30;"CE";AT 1,30;"DF"
;AT 1,11; INK 5; BRIGHT 1;"cross
fire"
  40 PRINT AT 5,8;"q";AT 5,10;"t
o";AT 5,13;"t";AT 5,20;"up"
  50 PRINT INK 6;AT 7,8;"a";AT 7
,10;"to";AT 7,13;"g";AT 7,20;"do
wn"
  60 PRINT INK 5;AT 9,8;"u";AT 9
,10;"or";AT 9,13;"o";AT 9,20;"le
ft"
  70 PRINT INK 4;AT 11,8;"i";AT
11,10;"or";AT 11,13;"p";AT 11,20
;"right"
  80 PRINT INK 3;AT 13,8;"b";AT
13,10;"to";AT 13,13;"space";AT 1
3,20;"fire"
  90 PRINT INK 5; BRIGHT 1;AT 15
,8;"h";AT 15,20;"halt"
 100 PRINT INK 5; BRIGHT 1;AT 17
,8;"s";AT 17,20;"halt";AT 17,25;
"off"
 110 PRINT INK 2;AT 19,8;"zero";
AT 19,20;"abort"
 130 FOR a=0 TO 1500: IF INKEY$=
"" THEN NEXT a: GO TO 1000
 140 RANDOMIZE USR 27903: INK 7:
BEEP .1,0: BEEP .1,3
 150 LET z$=FN x$(PEEK 23296)+FN
 x$(PEEK 23297)+FN x$(PEEK 23298
)
 160 FOR a=1 TO 6: IF z$<=b$(a)
THEN NEXT a: GO TO 200
 170 IF a<>6 THEN FOR b=6 TO a+1
 STEP -1: LET b$(b)=b$(b-1): LET
 a$(b)=a$(b-1): NEXT b
 180 GO TO 2000
 200 CLS : PRINT AT 0,8; INK 6;"
high{score{table": FOR a=1 TO 6:
 PRINT AT a*2+4,8; INK (8-a);a$(
a);AT a*2+4,18;b$(a): NEXT a
 210 PAUSE 200: GO TO 30
1000 RANDOMIZE USR 27900: BEEP .
1,0: BEEP .1,12: GO TO 30
2000 CLS : LET c$="{{{": PRINT A
T 3,4; INK 6;"well{done{your{ini
tials";AT 5,4; INK 5;"may{be{ent
ered{for{the";AT 7,9; INK 4;"his
core{table"
2010 FOR c=1 TO 3: PRINT INK 3;A
T 13,9;c$;AT 13,16;z$;AT 13,8+c;
 INK 2; FLASH 1;"{"
2020 LET d$=INKEY$: IF d$>="a" A
ND d$<="z" THEN LET c$(c)=d$: GO
 TO 2080
2030 IF d$>="0" AND d$<="9" THEN
 LET c$(c)="WXYZ[\]↑_£"(VAL d$+1
): GO TO 2080
2040 IF d$=" " THEN LET c$(c)="{
": GO TO 2080
2050 IF d$=CHR$ 12 AND c>1 THEN
LET c=c-2: LET c$(c+1 TO )="{{{"
: GO TO 2080
2060 GO TO 2020
2080 BEEP .1,30: NEXT c: LET a$(
a)=c$: LET b$(a)=z$: GO TO 200
```

Jason Charlesworth does it with an Amstrad seven pages on, but here he does it with a Spectrum.





Listing 2.

```
  10 DEF FN x(a$)=16*(CODE a$-48
-(39 AND a$(1)>"9"))+CODE a$(2)-
48-(39 AND a$(2)>"9")
  20 CLEAR 30199: LET x=30200: P
RINT "Program 2 poking in data""
"Please wait."
  30 FOR a=100 TO 400 STEP 10: R
EAD a$: LET t=256+FN x(a$( TO 2)
)+FN x(a$(3 TO 4)): LET a$=a$(5
TO ): IF LEN a$<>160 THEN GO TO
60
  40 FOR b=0 TO 79
  50 LET z=FN x(a$(b*2+1 TO b*2+
2)): POKE x,z: LET x=x+1: LET t=
t-z: NEXT b: IF NOT t THEN NEXT
a: PRINT "Finished,no errors.":
STOP
  60 PRINT "Error in line ";a: S
TOP
 100 DATA "2bdbc3e86dcdad6fcd1e6
fcd8372cde173cd2474cd9775cdf76fc
da670cd0a78cd696f3a075bfe00280f3
a085bfe00c0cd616efeff20da18d2cd7
f6efe0020c8cd1e6fcd0b73cde173cdf
974cd9775cdf76f"
 110 DATA "27b7cda670cd0a78cd696
f3a075bfe00280f3a085bfe00c0cd616
efeff20da18d2cd7f6efe0020c8cd1e6
fcdbb72cde173cda474cd9775cdf76fc
da670cd0a78cd696f3a075bfe00280f3
a085bfe00c0cd61"
 120 DATA "28ed6efeff20da18d2cd7
f6efe0020c8cd1e6fcd4772cd5d73cd8
```

(listing 2 continued on page 70)

Instead of ten aliens, Cl
for his Commodo

What happened nex



Saturday morning.

Waltzed into my local computer shop.

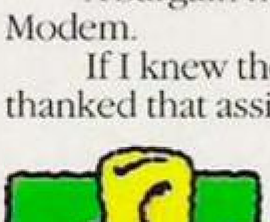
Packed as usual with masses of kids enjoying the arcade games.

Surely I could put my Commodore 64 to better use. Helpful assistant suggests a Commodore Communications Modem.

Tells me it comes with a year's free subscription to Compunet, a new network service, saving me a cool thirty quid.

A bargain not to be missed, so I bought a Modem.

If I knew then, what I know now, I'd have thanked that assistant more.



Saturday (one week later).

Fantastic.

My Compunet membership came through this morning.

Hurriedly plugged the Modem into my 64's cartridge port, and hooked up to the telephone line.

Can't wait.

At last I can communicate with other Commodore 64 Modem owners and giant mainframes.

What's more, I can also access databases throughout Europe and the U.S.A.*

This is what home computing's all about.



Sunday morning.

Raining.

Tapped in my Compunet I.D. and personal password.

Wow, what a directory!

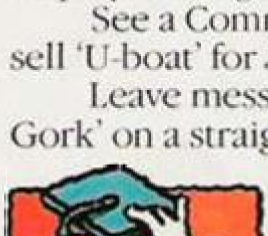
Decide to pit my wits against other Modem users by entering Multi User Dungeon, an interactive on-line game.

Should stretch the old grey matter a bit.

Then a quick look in 'The Jungle.' This is an open area where other Modem owners display messages.

See a Commodore user in Fife wants to sell 'U-boat' for £3.00.

Leave message offering him 'Mighty Gork' on a straight swap.



Monday evening.

Move on to the Compunet Software Park.

What a choice. Loads of high quality bargain programs.

Particularly interested in educational software, so I call up 'The Study.'

Download free physics package to help with my exams.

Clive bought a Modem lore 64.



ext changed his life.



Tuesday evening.

Dad's turn.
I don't get a look in as he's busy teleshopping.

Actually it's amazing what bargains turn up. He even finds a new house.

Mum said she doesn't want to move and anyway his dinner's getting cold.

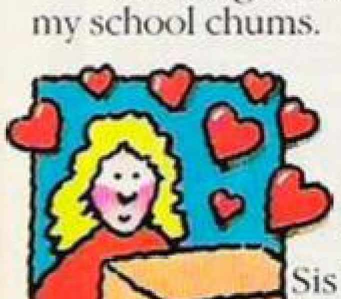


Wednesday evening.

Discover I can join BLAISE,* the computer service for the British Library.

Their catalogue of books dates way back to 1950.

Should give me an interesting edge over my school chums.



Thursday evening.

Sis has a go.
She keys into Prestel.*
Imagine, over 300,000 pages of information and news.

What does she choose? The lonely hearts section.

She's disappointed. Couldn't find Simon le Bon's private number.



Friday evening.

Yippee! Receive a reply from the guy in Fife.

He fancies taking on Gork.

What's more he's written a program he'd like my opinion on.

He transfers it direct, using the free user to user software.

I've made my first computer pal.

What a week. Best one I've had since getting my Commodore 64.

Sure am glad I got the Modem instead of all those aliens.

The Commodore 64 Communications Modem comes as a complete package with a year's free subscription to Compunet, for just £99.99 inc. VAT.

Find out how a Modem can change your life. See it now at Spectrum, Curry's, Comet and selected Commodore dealers.

commodore

For further information phone or write to:
Commodore Communications Modem, 1 Hunters Road, Weldon, Corby, Northamptonshire NN17 1QX. Tel: 0536 205252.

*Require additional subscriber charges. Prestel is a reg.d.trademark of British Telecom.

(listing 2 continued from page 68)

```
273cd9775cdf76fcda670cd0a78cd696
f3a075bfe00280f3a085bfe00c0cd616
efeff20da18d2cd7f6efe0020c8cd337
2c3026dcdad6f06"
 130 DATA "27a002c5cd1e6fcd8372c
de1730600c5cd2474cd596efe002852c
110f2cd1e6fcd0b73cde1730600c5cdf
974cd596efe002839c110f2cd1e6fcdb
b72cde1730600c5cda474cd596efe002
820c110f2cd1e6f"
 140 DATA "231ecd4772cd5d730600c
5cd8273cd596efe002807c110f2c1109
8c9c1c1c901feefed78e601c93a625b0
610e603fe03280206082115b7efeffc
0232323232310f53effc921035b7e23b
623b6fe00c83a06"
 150 DATA "185f5bd6012732065b3a6
25b3d32625b3e04c997328d5ccd6b0d2
1090722885c21277f227b5c3e45328f5
c21885c110c6f060d1afe20200334180
3cdaa791310f2210b0922885c06051ac
daa791310f921d7"
 160 DATA "184e7e227b5c3e1232885
c3a065b47e6f01f1f1f1fcdaa7978e60
fcdaa7921377e3e46328f5c017458cde
f7906647610fdc906041320110400031
820050e110b0415040b3a625b3c32625
b3a065bc6012732"
 170 DATA "20d9065b21035b9777237
72377cd9d6ecd727821d77e227b5c3a6
25bc6104721035b3a065b111916ed538
85ccd6e7910f7cd567621635b0620360
02310fbc901febfed78e610fe10280b0
1fefded78e602fe"
 180 DATA "1ac30228f801feefed78e
601fe0128053e0132085b210a5b7efe0
6d0237efe06d02323237efe06d0237ef
e06d03e0132085bc921005b977723772
37732085b32065b3d32625bcd3372c90
000000000000000"
 190 DATA "1ebd00000000000000000
0000000cd7278cd8372cde173cd3372c
d5676cd0a78cd2474cd9775cdf76fcdf
76fc901fe7fed78e61ffe1fc811095b2
16b5b131afe06381806087efe0028052
310f8180c1b1a77"
 200 DATA "17254f2313061070cd867
013217b5b1a13fe06381606087efe002
8052310f8180a0eef711a237747cd867
01321635b131afe06381806087efe002
8052310f8180c1b1a774f1323069f70c
d86701321735b1a"
 210 DATA "1ef3fe06d806087efe002
8042310f8c90e107123131a7747cd867
0c9cd58797c0f0f0fe603f6586736470
1fe003e102e03ed79ee102d20f997ed7
9c921635b06107ec52323fe0028094f2
b4623e5cd5879e1"
 220 DATA "1e17c110ec21635b11fc0
0cde470216b5b110400cde47021735b1
10004cde470217b5b1100fccde470c90
604c5d5e57efe00283e4f23467b805f7
a8157fe063003110000fefa380311000
078fe0830031100"
 230 DATA "21d600fca838031100007
32b727bb2280e4a43cd4f7cfe002805e
1cd3471e5cd8670e12323d1c110b3c9c
5e51e08dd21115b3a625be603fe03286
bdd7e03fe7b285479dd9600fe10304c7
8dd9601fe0c3044"
 240 DATA "1f0be136002336002be5c
d5879dd4e00dd4601dd7e04dd36037bd
d3604[illegible]3cd5a753e03cd8477210716228
85c21d77e227b5c21005b3a625b3ccd6
e793a065b57cd0072e1c1c9dd23dd23d
d23dd23dd231d20"
 250 DATA "227698e1c1c91e10dd7e0
3fe7b283779dd9600fe083028f78dd960
1fe083027e1e597772377cd5879dd4e0
0dd4601dd36037bdd360403211f7e3a6
25be60417856fcd9d7b188bdd23dd23d
d23dd23dd231d20"
 260 DATA "1d78b5e1c1c921035b7e2
3b6fe00281d237e92277730082b7ed60
1277738f821191622885c21035b97cd6
e79c9237efe00c8ba30dc360018e6213
f7211095b010800edb0c98008f75780a
0085711115b0610"
 270 DATA "1c3dcdf672783d87874f8
781c61e1213783de60c171717c628121
3783de60c1f1fe6023d87121312137de
60720043a785c3ce607121310cac9111
15b0608cdf672783d878787c628121
37de65fc62d1213"
 280 DATA "1d3c7de60f12137ce6031
2137de60720053a785cf601e6071717f
602121310cec911115b0608cdf672783
de60787878780c62812137de65fc61
412133efd12137cade6023d12137d84e
60720033a765ce6"
 290 DATA "2012071717f602121310c
bc9d52a765ced5b785c1916011911e10
61922765cd1c9dd21115b0608cdf6727
83d17171717c628dd770084e65fc61ed
d77017de6015f87878383d603dd77027
cade604d602dd77"
 300 DATA "1f6b037d84e60720043a7
85c3ce6071717f602dd7704dd23dd23d
d23dd23dd2310b5c921115b0610c54e2
3462323237e328f5c23e5211f7e3a625
be60417856fcd9d7be1c110e1c921115
b0610c5cd9473c1"
 310 DATA "1fe6232323232310f4c94
e2346235e2356237ecdfc73e5328f5c2
11f7e3a625be60417856fcd9d7b7882f
e0a3003781602fe9e380037816fe47798
3fe0a3003791e02feee3803791efe4fc
d9d7be12b722b73"
 320 DATA "21ee2b702b71c90608211
15bc5e54e23462323237ecd5a75e1c12
32323232310ebc9e5083e7bba201e08f
eff2812f5c53d773ccd8477c1f13dfef
f2803cd8477e12b2b2b2bc1c908e1c92
1115b0608c5cd36"
 330 DATA "1d7974c123232323232310f
4c94e2346235e2356237ecdfc73cd5a7
5e6fc577ee6033cfe03200197b2772b5
623cb4228167ad6028780fe0a3003781
603fe993803701001471015157a07821
481fe0a30037916"
 340 DATA "1f2902fee53803791600 4
f7b1d200e3a785ce603573a785ce607c
6085f7ecd5a752b722b732b702b71c92
1115b0608c5cdb674c1232323232310f
4c94e2346235e2356237ecdfc73cd5a7
5e6fc577ee6033c"
 350 DATA "1efffe03200197b2772b5
6237882fe0a30023e98fe99380023e094
77983fe0a30023ee74f7ecd5a752b2b2
b702b71c921115b0608c5cd0b75c1232
32323232310f4c94e2346235e2356237ec
dfc73cd5a75e6fc"
 360 DATA "21d8577ee6033cfe03200
197b2772b567882fe0a30031602 78fe9
a380316fe78477983fe0a30031e0379f
ee638031efd794f237e2bcd5a75722b7
32b702b71c9e5d5c5f5e6fc1f1f328f5
cf1f5e603171717"
 370 DATA "1c845f17835f1600216f7
c193a625bcb57280411d800 19e603171
717175f191919191f5f19cdca7af1c1d
1e1c9cd1377cd227621095b4e23463e0
6b83018790282fe103811fef0300d4f0
807cd4f7cfe8020"
 380 DATA "1d38032b7123234e23463
e06b93018788383fe10380ffea0300b4
70ef8cd4f7cfe80200170234e23463e0
6b83018798282fe103811fef0300d06a
84fcd4f7cfe802003 2b7123234e23467
8fe06381583 83fe"
 390 DATA "23c110380ffea0300b0e0
747cd4f7cfe80200170cd5676c901fef
b110000ed78e61ffe1f28021eff01fef
ded78e61ffe1f28021e0101fedfed78e
605fe05280216 01ed78e60afe0ac816f
fc9ed4b095b78fe"
 400 DATA "1f4a06300bfe002822060
8cd8477181bcdde777bfe002813cdde7
779d6084f0604ed43095b780608cd847
7ed4b0b5b79fe06300bfe0028220ee3c
d8477181bcdc2777bfe002813cdc2777
8d608470e04ed43"
```

Listing 3.

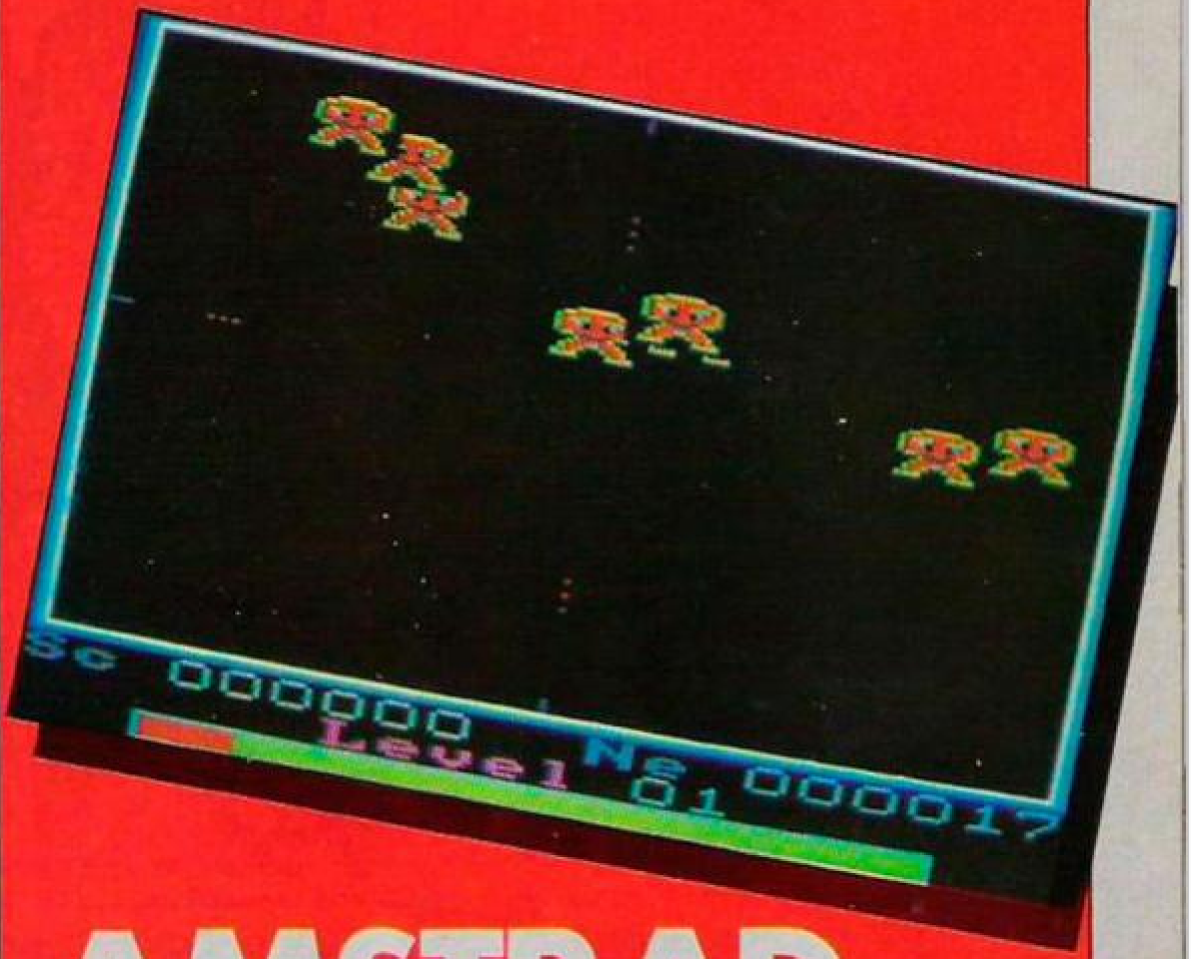
```
 10 DEF FN x(a$)=16*(CODE a$-48
-(39 AND a$(1)>"9"))+CODE a$(2)-
48-(39 AND a$(2)>"9")
 20 CLEAR 30299: LET x=30300: P
RINT "Program 3 poking in data"'
"Please wait."
 30 FOR a=100 TO 390 STEP 10: R
EAD a$: LET t=256*FN x(a$( TO 2)
)+FN x(a$(3 TO 4)): LET a$=a$(5
TO ): IF LEN a$<>160 THEN GO TO
60
 40 FOR b=0 TO 79
 50 LET z=FN x(a$(b*2+1 TO b*2+
2)): POKE x,z: LET x=x+1: LET t=
t-z: NEXT b: IF NOT t THEN NEXT
a: PRINT "Finished,no errors": S
TOP
 60 PRINT "Error in line ";a: S
TOP
 100 DATA "21230b5b790ee8cd8477e
d4b0d5b78fe06300bfe0028220697cd8
477181bcdde777bfe002813cdde7779d
6084f0604ed430d5b780697cd8477ed4
b0f5b79fe06300bfe0028220e08cd847
7181bcdc2777bfe"
 110 DATA "1dc7002813cdc27778d60
8470e04ed430f5b790e08cd8477c9ed4
b095b78fe063010fe00280f06083d320
a5b3ccd84771803cdde77ed4b0b5b79f
e063010fe00280f0ee83d320b5b3ccd8
4771803cdc277ed"
 120 DATA "1c464b0d5b78fe063010f
e00280f06973d320e5b3ccd84771803c
dde77ed4b0f5b79fe063010fe00280f0
e083d320f5b3ccd84771803cdc277c92
1b3775f3e04935f16001919195e23562
37eeb328f5ccdef"
 130 DATA "1e1d797dfe37c821fa001
71717175f16001911 0600cdb503c9577
e47777e06977e03b77e02377e46c5cd6
67c975fbe20021e013effae777c0f0f0
fe603f6586736 07c1c9c5cd667c473e8
02604e71f10fd06"
 140 DATA "1b0f084f1e007ea120021
e017ea97724 10f4257c0f0f0fe603f65
8673607c1c93a785cfe32d8d63232785
ccd1978c93a075b3d32075ba71717c60
84f06bb1e04cd587904cd58790 50c1d2
0f421fa00060009"
 150 DATA "1e15110400cdb503c9212
77f227b5c21637847fe0028087e23fef
f20fa10f87e23feffc8cdaa7918f6120
20e1104ff0bff0d04040 30403ff97328
d5ccd6b0dcd32790604cd0e7906a8cd0
e793e4732005821"
 160 DATA "1a32071622885c21d77e2
27b5c21005b97cd6e7921191622885c2
1035bcd6e79210f1622885c3a065be6f
01f1f1f1fcdaa793a065be60fcdaa792
1277f227b5c21011622885c97cd43783
e012 10e1622885c"
 170 DATA "1d7dcd43783e022112162
2885ccd43782 1e15a3e4277237723770
61b23364410fb3e3c32075bf5cd1978f
13d20f83e3c32075bc90e08cd667c36f
f2436ff2436ff2436ff7c0f0f0fe603f
65867364779c603"
 180 DATA "222c4ffef838dfc906040
e00cd667ce5111f001936f07c0f0f0fe
603f6586736472 3364 7e1360f0478fea
c20dfc9c5cd667cfe00280a473e801f1
0fdae77c1c93e8018f8e5d5c5f5e51e0
32323862777300e"
 190 DATA "25c32b7ec601277730062
b7ec6012777e13e45328f5c7ef5e6f01
f1f1f1fcdaa79f1e60fcdaa79231d20e
bf1c1d1e1c9e5d5c5f5ed4b885c0ced4
3885c0da7cb10cb10cb10cb11cb11cb1
1171717ed5b7b5c"
 200 DATA "2285833001145fcd667c0
6081a7713 2410fa257c0f0f0fe603f65
8673a8f5c77f1c1d1e1c9e5d5c5f5e5d
de178cdb122ca727aed434d7a322a7ae
57c0f0f0fe603f658673a8f5c7723772
37711200019772b"
 210 DATA "1c41772b77e13e1008dd5
e00dd5610 0603afcb1bcb1a1f10f9232
3ae772b7eaa772b7eab77083d281908d
d23247ce60720d701432d085f083e109
38047cd667c18c77c0f0f0fe603f6586
73a8f5c77237723"
 220 DATA "1f8877f1c1d1e1c9ed43b
07ae57c0f0f0fe603f658673a8f5c772
37711200019772b77e13e1008dd5e00d
d5610237eaa772b7eab77083d281908d
d23247ce60720e5085f080148673e109
38047cd667c18d5"
 230 DATA "1e867c0f0f0fe603f6586
73a8f5c18a0e5dde178cdb122ca457be
d43247b32017be57c0f0f0fe603f6586
73a8f5c7723772377112000 19772b772
b77e13e0c08dd5e00dd560c0604afcb1
bcb1a1f10f92323"
 240 DATA "1e57ae772b7eaa772b7ea
b77083d281908dd23247ce60720d701b
e9a085f083e0c938047cd667c18c77c0
f0f0fe603f658673a8f5c7723772377c
9ed43837be57c0f0f0fe603f658673a8
f5c772377112000"
 250 DATA "220519772b77e13e0c08d
d5e00dd560c237eaa772b7eab77083d2
81908dd23247ce60720e5085f0801a88
e3e0c938047cd667c18d57c0f0f0fe60
3f658673a8f5c18a4e5d5c5f5e5de17
8cdb122ca097ced"
 260 DATA "1db343e67b32ca7be57c0
f0f0fe603f658673a8f5c772377e13e0
808dd5e00 0604afcb1b1f10fb23ae772
b7eab77083d281908dd23247ce60720e
1012c78085f083e08938047cd667c18d
17c0f0f0fe603f6"
 270 DATA "200058673a8f5c772377f
1c1d1e1c9ed43357ce57c0f0f0fe603f
658673a8f5c77e13e0808dd7e00ae770
83d281908dd23247ce60720ee085f080
120783e08938047cd667c18de7c0f0f0
fe503f658673a8f"
 280 DATA "24ac5c18b4e5c5cd667cf
e002808477ea71710fc18017ee680c1e
1c978c3b122000000000000101010f3f7
fc9c97f3f0f01f08080f0fcfe2727fef
cf0800301010f3f7fe4e47f3f0f01c08
000f0cfe9393fe"
 290 DATA "21a6fcf0800f01010f3f7
fd2d27f3f0f01808080f0fcfe4b4bfef
cf080003078 7cfefffffe7c783000000
c1e3e7fffff7f3e1e0c0000183c7e7ef
fff7e7e3c180000183c7e7effff7e7e3
c18000001070f1f"
 300 DATA "20ecffff1f0f070100000
0e0f0f8ffffff8f0e08000003d707cf9f
cfd787d381e000080700cc1002700cc3
08000001d7c7df8fcf9787d3c1a00000
0d00cc10827004d308000001a7d7cfaf
cf9787d3c1a0000"
 310 DATA "19da00a63000508240930c
08000001010303 0f7e7e0f03030101808
0c0c0f07e7ef0c0c080800010103030f1
e1e0f03030101 8080c0c0f07070f0c0c
080800101030 30f06060f03030101808
0c0c0f06060f0c0"
 320 DATA "23b4c08080030e0e1f3f2
7273f1f0e0e03c07070f8fce4e4fcf87
070c0030f0f1723373f3f1d080d03c0b
010b0fcfcecc4e8f0f0c0030f0b113b3
f3f3b110b0f03c0f0d088dcfcfcdc88d
0f0c0ff013f4080"
 330 DATA "270c90a1a37fbf927ffe0
000c2fae7fac2000000801f013f40809
0a1a37fbf927ff00000c1fae7fac4000
000800 3013f408090a1a37fbf927f800
000c4fae7fac1000000080187e7ee7e77
e7e187e9999e7e7"
 340 DATA "165199997e3c1899ffff9
9183c010107191120 20f8f8202011190
7010180 80e0988804041f1f04048898e
080800000000000000061d3b3b3f1f0d0
0000000000000000f0e8dcfcfcf8f00
000000000000000d"
 350 DATA "18c63b3b7f7f3f3b1c070
0000000000000f0ececfefefcfcf8600
000000000 01070f1f1860e1e0700d070
3000000000 80e0f0f006c3e30f00b0e0c
00000000001070e180960ccc06400060
30000000080e070"
 360 DATA "1882300643e306c00060c
000003c7e6666666667e3c183818181 81
83c3c7c7e063e7c607e7e7e7f030f0f0
37f7e0e1e36667f7f06067f7f607e7f0
37f7e3f7f607e7f637f3e7f7f06060c0
c18183e7f637f7f"
 370 DATA "1f15637f3e3e7f637f3f0
37f7e3e7f637f7f6363637e7f637e7e6
37f7e3f7f6060 60607f3f7e7f6363636
37f7e3f7f607878607f3f3f7f6078786
060603f7f6060637f3f0363636 37f7f6
363637f7f181818"
 380 DATA "1f4f187f7f7f7f0606666
67e3c63666c78786c666360606060606
07f3f777f6b63636363636363737b6f6
763633e7f63636363 7f3e7e7f63637f7
e60603e7f63637b7f3e037e7f637f7e6
c66633f7f607e3f"
 390 DATA "12d3037f7e7f7f0c0c0c0
c0c0c636363636363 7f3e636363636 33
61c0863636363636b7f776363 63361c1c3
66363636363 7f3f037f7e7f7f0e1c387
07f7f000000000000000000000000000
000000000000000"
```


BRIAN JACKS
SUPERSTAR
Challenge
"Winning's not the most important thing – it's the only thing"
Brian Jacks, world famous Judo and TV Superstars champion made it very clear.
"It better be good," he said.
We didn't feel like arguing the point, so we just got to work.
The result is one of the most exciting, addictive and challenging games you will ever have attempted.
We feel it only fair to warn you though – challenge Brian Jacks to anything and YOU'D better be good!
Brian Jacks challenges you to these eight events:
★ ARM DIPS ★ SQUAT THRUSTS
★ CANOEING ★ CYCLING
★ FOOTBALL ★ SWIMMING
★ ARCHERY ★ 100 METRES
Available from all leading software retailers
OR order today by mail.
CBM-64 £8·95 fast load cassette £12·95 disc
48K Spectrum/BBC-B/Electron/Amstrad £7·95
prices include VAT & p/p
martech
Martech is the registered trade mark of Software Communications Limited
Martech House, Bay Terrace, Pevensey Bay, East Sussex BN24 6EE
TRADE ENQUIRIES WELCOME. PHONE: (0323) 768456 TELEX: 87465 EXIM Brighton
48K SPECTRUM · COMMODORE 64 · BBC 'B' · ELECTRON · AMSTRAD

AMSTRAD

CROSS FIRE

The game where the most dangerous thing on the screen is you. Jason Charlesworth with a froody freaker for the CPC464.



CROSS FIRE is a totally new game which should zap the mega zappers, freak out the froodies and generally blow your mind — and hopefully several alien nasties. The game follows the normal shoot-'em-up idea of zap, maim or destroy everything in sight and as such requires fast reflexes.

But what do the aliens do while all this is going on? Do they fire back with super deadly plasma bolts? Do they mercilessly hunt you down? Or do they set hideous traps for you? No, all they do is continue wandering round the screen blundering into your bases.

With no enemies this may sound like a simple game but unfortunately you do have something far worse to contend with — yourself. Not only can you shoot aliens, you can also shoot holes in your tracks or even worse, you can shoot yourself.

This is all due to the layout of your bases. These are mounted on tracks on each of the four sides of the screen and the controls move each pair of bases simultaneously.

That means you cannot just move your left laser without also moving the right as well. The fire button works on all four lasers at once. However if you shoot an opposite track a hole forms in it and your laser on that track cannot go past it.

The object of the game is to shoot all the aliens, there are eight screens — Space Invaders, Balls, Helicopters, centipedes, Dumbells, Ultra friendly aliens, Suitcases and a Scottish snake. All these move in smooth pixel graphics.

On each screen eight of 16 aliens appear — depending on the round — though when you have wiped out that batch, another batch appears.

Each alien shot earns you a number of points corresponding to the level you are on and this number of points is deducted from score titled "Needed". This amount must be reduced to zero before you are allowed on to the next screen and on each new screen, the amount of points needed increases.

If you do not get enough points within the time limit — the bar at the bottom of the screen counts off 60 seconds — you restart the same screen. You can only go onto the next screen when you have lasted the 60 seconds and reduced the "needed" score to zero.

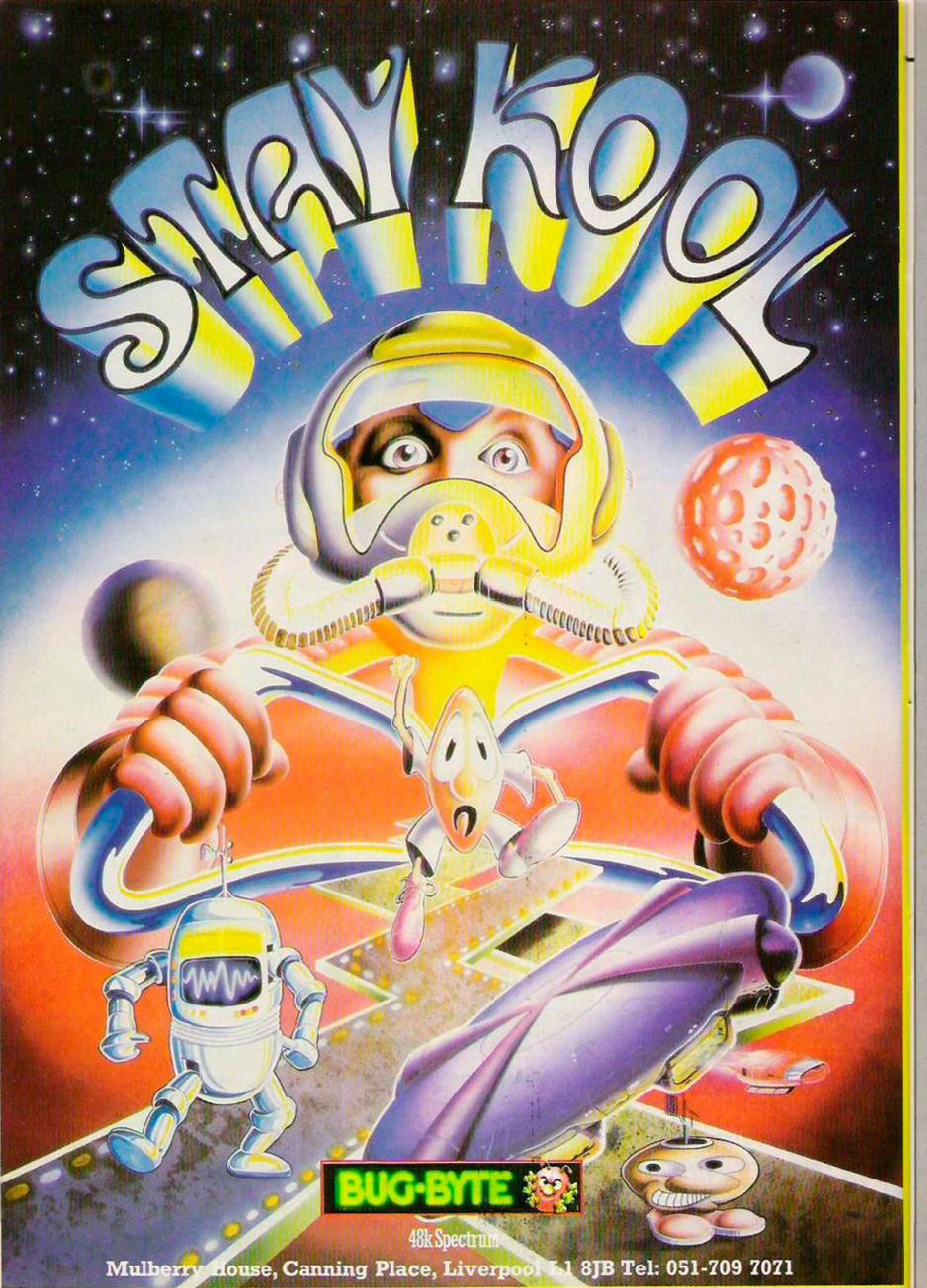
The game ends when all four bases have

(continued on page 75)

Listing 1.

```
100 MEMORY 32767
110 DEF FN x(n)=10*INT((PEEK(n)/16))+PEEK(n)-16*INT(PE
EK(n)/16)
120 BORDER 0:INK 0,0:PAPER 0:CLS:LOAD ""
121 POKE 34257,196
130 DIM N$(5):DIM N(5):FOR A=1 TO 5:READ N$(A),N(A):NEXT A
140 GOSUB 320
150 GOSUB 210
160 GOSUB 180
170 GOTO 140
180 MODE 1:PEN 3:PRINT,"Cross Fire"
190 FOR a=1 TO 5: LOCATE 5,2*a+5:PEN 2:PRINT n$(a):LOC
ATE 30,2*a+5:PRINT n(a):NEXT a
200 LOCATE 10,25:PRINT"Press any key please":IF INKEY$
="" THEN 200 ELSE RETURN
210 INK 0,0:INK 1,0:INK 5,24:INK 8,15:INK 9,16:INK 11,
7:INK 13,3:INK 14,8:INK 15,1
220 ENV 1,5,3,1,1,0,20,15,-1,2
230 ENV 2,5,3,2,5,-1,1,5,-2,3
240 CALL 34000:sc=FN x(37328)+100*FN x(37329)+10000*FN
 x(37330)
250 FOR a=1 TO 100:a$=INKEY$:NEXT a
260 FOR a=1 TO 5:IF sc<n(a) THEN NEXT a:GOTO 300
```

(listing 1 continued on page 75)


STAY KOOL
BUG-BYTE
48k Spectrum
Mulberry House, Canning Place, Liverpool L1 8JB Tel: 051-709 7071

(continued from page 73)

been destroyed — you do, however, get four new bases every four rounds. If you want to see each of the levels, don't press any key when it lists out the keys and after about 10 seconds it gives a demo mode. To get out of this mode press 0.

To type in the game type in listing 1 and save it to tape with

SAVE "CROSS FIRE"

Next type in listing 2 and run it. This Pokes in all 5K of code and checks it. If it finds an error it will tell you which lines the error could be in. It should be noted that the program only uses a simple checksum and so care should still be taken when entering the code as it cannot detect every error. When the program gives a "no errors found" message, save the code directly after Cross Fire with

SAVE "CODE",B,34000,5000

The game should now be ready. To run the game rewind the cassette and type

RUN

Finally a few tips for playing the game. At the start of the game you should move your base very rapidly otherwise you may find the aliens wipe some out. When you fire, if you move the base whilst firing, it spreads the bullets out and so they are more likely to hit an alien. Perhaps the most important tip is to turn the sound down otherwise the constant tick of the clock may drive you mad.

(listing 1 continued from page 73)

```
270 IF a<>5 THEN FOR b=5 TO a+1 STEP -1:n$(b)=n$(b-1):
n(b)=n(b-1):NEXT b
280 FOR b=0 TO 1000:NEXT b:PEN 3:MODE 1:PRINT"Please e
nter your name for the high      score table.":INPUT n$
(a)
290 n(a)=sc
300 GOSUB 180
310 GOTO 140
320 MODE 1: PEN 3:PRINT"      Cross Fire By J.Charleswo
rth":PEN 2:PRINT:PRINT:PRINT,"a          Up":PRINT:PRINT,
"z          Down"
330 PRINT:PRINT,"/          Right":PRINT:PRINT,".        L
eft":PRINT:PRINT,"SPACE      Fire"
340 PRINT:PRINT,"0          Abort":PRINT:PRINT,"h        H
old":PRINT:PRINT,"s          Start (from hold)"
350 PRINT:PRINT:PRINT"      Press J for Joystick or K for
 Keys"
360 FOR a=1 TO 3000:a$=INKEY$:IF a$="J" OR a$="j" THEN
 380
370 IF a$<>"k" AND a$<>"K" THEN NEXT a:GOTO 390 ELSE P
OKE 34766,47:POKE 36124,69:POKE 36134,71:POKE 36151,31
:POKE 36142,30:RETURN
380 POKE 34766,76:POKE 36124,72::POKE 36134,73:POKE 36
151,74:POKE 36142,75:RETURN
390 INK 0,0:INK 1,0:INK 5,24:INK 8,15:INK 9,16:INK 11,
7:INK 13,3:INK 14,8:INK 15,1
400 ENV 1,5,3,1,1,0,20,15,-1,2
410 ENV 2,5,3,2,5,-1,1,5,-2,3
420 CALL 34288:GOTO 320
430 DATA Danger Mouse,500,Penfold,400,The Bat out of h
ell,300,Shaggy the Yak,200,Albert Aadvark,100
```

Listing 2.

```
10 MEMORY 32767:RESTORE:DIM t(62):FOR a=0 TO 62:READ T(A)
:NEXT A
20 FOR a=0 TO 62:READ a$:t=0:FOR b=0 TO 79:c=VAL("&"+MID$
(a$,b*2+1,2)):t=t+c:POKE 34000+80*a+b,c:NEXT b
30 IF t<>t(a) THEN PRINT"Error in lines 100-130 or line "
;130+a*10:END
40 NEXT a
50 PRINT"Finished,no errors":END
100 DATA 12116,11842,11499,8483,10699,8515,7608,7750,7593
,8128,5707,8005,7017,8963,9596,8113,9136,8540,10051,7272,
8056,9082,7433,8723,8471
110 DATA 7387,7274,8647,7804,8967,8355,10056,7606,6253,80
20,8863,11218,9006,8883,9227,10895,6615,957,2999,12527,11
656,10811,10761,11740,11692
120 DATA 10463,9077,17547,15439,14468,12867,13847,13898,1
5264,15198,11410,5904,0
130 DATA CDB687CD1287CDE789CDB985CD1C8ACD8E8CCDCD87CD5988
CDDA8ECD75873AD791FE00280F3AD891FE00C0CD6B86FEFF20DA18D2C
D8986FE0020C8CD1287CD3E8CCDB985CDC688CD8E8CCDCD87CD5988
140 DATA CDDA8ECD75873AD791FE00280F3AD891FE00C0CD6B86FEFF
20DA18D2CD8986FE0020C8CD1287CDA088CDB985CD4388CD8E8CCDCD8
7CD5988CDDA8ECD75873AD791FE00280F3AD891FE00C0CD6B86FEFF
150 DATA 20DA18D2CD8986FE0020C8CD1287CDA08ACDB985CDEE8ACD
8E8CCDCD87CD5988CDDA8ECD75873AD791FE00280F3AD891FE00C0CD6
B86FEFF20DA18D2CD8986FE0020C8CD9F8DC3D3843E1CF56F260054
160 DATA 1E040E04CD1DBC064C36002310FBF13CFEC320E7CDB38D21
0000110000CD10BDC921E785C3AABC0202006401140F1400CDB687060
2C5CD1287CDE7890600C5CD1C8ACDDA8ECD6186FE002852C110EFCD
170 DATA 1287CD3E8C0600C5CDC688CDDA8ECD6186FE002839C110EF
CD1287CDA0880600C5CD4388CDDA8ECD6186FE002820C110EFCD1287C
DA08A0600C5CDEE8ACDDA8ECD6186FE002807C110EFC11098C9C1C1
180 DATA C93E20CD1EBB3E00C03CC93A45920610E603FE0320020608
21E5917EFE00C0232323232310F53EFFC921D3917E23B623B6FE00C83
AD691D6012732D6913A45923D3245923E04C997CD0EBC3E0CCD90BB
190 DATA 21020522439111FB86060E1ACDBE901310F9210306224391
0609110987 1ACDBE901310F93AD69147E6F01F1F1F1FC630CDBE9078E
60FC630CDBE903E0F06000E000D20FD0520F83D20F3C95072657061
200 DATA 726520746F206469654F6E206C6576656C203A45923CFE09
20023E013245923AD691C6012732D69121E1910662360023 10FB21D39
1977723772377CDA786CD1A8F3A4592C6104721D3913AD691110E16
210 DATA ED534391CD969110F7CDB38D21469206203600 2310FB2100
0011 0000CD10BD3E3C32D791C93E2CCD1EBB28103E3CCD1EBB28F9210
000110000CD10BD3E20CD1EBB28053E0132D89121DA917EFE06D023
220 DATA 7EFE06D02323237EFE06D0237EFE06D03E0132D891C921D0
9197772336002377 32D89132D691324592CD9F8DC93E4CCD1EBBC811D
991214E92131AFE06381806087EFE00280523 10F8180C1B1A774F23
230 DATA 13060870CD55881321 5E921A13FE06381606087EFE002805
2310F8180A0E48711A237747CD558813214692131AFE063815 06087EF
E002805 2310F818091B1A7723134F06A6701321 56921AFE06D80608
240 DATA 7EFE002804 2310FBC90E047123131A7747CD5588C9CD6791
C921469206107EC52323FE0028094F2846 23E5CD8091E1C110EC21469
211FC00CD9788214E9211 0400CD97882156921 10001CD97882 15E92
250 DATA 1100FFCD9788C90604C5D5E57EFE00285C4F23467B805F7A
8157FE01200B0E0143CD8091110000182CFE4E200B0E4E43CD8091110
000181D78FE06300B06034ACD8091110000180DFEA83809 06AC4ACD
260 DATA 8091110000732B727BB228114A43CD4591FE802005E1CD05
89E5CD5588E12323D1C11095C9C5E51E08DD21E1913A4592E603FE032
052DD7E04FE80203B79DD9600FE0830337 8DD9601FE10302BE13600
270 DATA 2336002BE5CD8091DD36040421031622439121D0913AD691
CD96913AD69157CDB489CDE185E1C1C9DD23DD23DD23DD23DD231D20B
1E1D1C91E10DD7E04FE80203379DD9600FE05302B78DD9601FE0830
280 DATA 23E1E5977723 77DD4E00DD4601DD3604043E45B93004DD36
00453E9AB83003DD770118 98DD23DD23DD23DD23DD231D20B9E1C1C92
1D5917E2BB6FE00281D2B7E9227773000237ED6012777 30F8210E16
290 DATA 2243919721D391CD9691C92B7EFE00C8DA30DC360018E621
E1910600C5CDF409C110F9C9 7B3D8787 87C60A77237887 8787C60A772
33A66923C3266925FE60377237 8E61C1F1F7723360023C921E19106
300 DATA 00C5E5CD308AE123232323 23C110F2C94E2346235E235623
7ECD0E8C78FE0020010 5FE01200 10CFE022 00104FE032000 10D1578FE0
4CC7A8AFE9CCC7A8A79FE02CC838AFE45CC838A7AFE00CC8D8A2B72
310 DATA 2B732B702B71CD70 0CC9F53E02935FF1160AC9F53E03933C
5FF1160AC9E5D5CD0D8D7DD1E157E6035F7AE60FC6025 7C9061021E19
1C5CDAD8AC110F9C9783DE60 3878787C610777 83DE6FC8787C61423
320 DATA 772378 3DE6FCFE002004 06FF0EFFFE04200406FF0E01FE08
20040601 0EFFFE0C20040601 0E01712370233600 23C9061021E191C5E
5CD028 8E12323232323C110F2C94E2346235E2356237ECD0E8C7983
330 DATA 4FFE022004FE49 200397935F788247FE042004FEA4200397
92572B722B732B702B71217C973A4592FE0420032 1A497CD9090C9060
821E191E5C5CD578 8C1E12323232323 10F2C94E2346235E2356237E
340 DATA CD0E8CE5793DFE01200BCD708C21FC96CD61903E464F7882
FE04200821FC96CD61903E9AFE9C200821FC96CD61903E0447CD708CE
136002B722B732B702B71C921E1910600C5788787877723C6107723
350 DATA 3600233A66923C326692E6023D77233680023C110E0C921E1
910600C5CDD88BC12323232323 10F4C94E2346235E2356237ECD0E8C7
9834FFE022004FE462004 3E00935F788247FE042004FE9C20043E00
360 DATA 9257CD708C2B722B732B702B71C9FE80C8FE00280E3D773C
2B2B2B2BE5CD748EE1C1C92B2B2B2BC1C9D52A66922 9292929292 9ED5
B66921911290019D1C921E1910600E5C5EB2A66922C226692EB7887
370 DATA 8787787 2377237BE6023D77237BE60117 3D772336 80C1E1
2323232323 10D4C9E5D5F53A4592FE0438013D1E003DA71FCB1B57217
C9219CD6190F1D1E1C9CD3E8DCD188D21D9914E23463E06B8301779
380 DATA 83FE063811FE4A300D4F0603CD4591FE80200 32B7123234E
23463E06B93016788282FE0E300FFEA43008 470E4ECD4591FE802001 7
0234E23463E06B8301779 83FE063811FE4A300D06AC4FCD4591FE80
390 DATA 20032B7123234E234679FE06381678 8282FE0E300FFEA430
080E0147CD4591FE80200170CDB38DC911 00003E48CD1EBB2803 1518 0
83E49CD1EBB2801143E48CD1EBB28021CC93E4ACD1EBBC81DC9ED4B
400 DATA D99178FE06300AFE00280B3D32DA911805 0604CDB68EED4B
DB9179FE06300AFE00280B3D32DB911805 0E4CCD878EED4BDD9178FE0
6300AFE00280B3D32DE91180506A8CDB68EED4BDF9179FE06300AFE
410 DATA 00280B3D32DF911805 0E02CD878EC901000021A88D11D991
ED80C9280C4C5028A80A50ED4BD99178FE06300BFE0028240604CD748
E181D0604CDB68E7BFE00 2813CDB68E79D6044F0604ED43D9917806
420 DATA 04CD748EED4BDB9179FE06300BFE0028220E46CD748E181B
CD878E78FE002813CD878E78D60 8470E04ED43DB91790E46CD748EED4
BDD9178FE06300BFE00282 2069CCD748E181BCDB68E7BFE002813CD
430 DATA B68E79D6044F0604ED43DD9178069CCD748EED4BDF9179FE
06300BFE0028240E02CD748E181D0E02CD878E78FE002813CD878E78D
```

(listing 2 continued on next page)

(listing 2 continued from previous page)

```
608470E04ED43DF91790E02CD748EC95F3E0493A71F1E00CB1B5721
440 DATA 7C9519CD6190C9C579875F1600623EC7906FCD1DBC1E00CD
A68EEE3C7723CDA68EEE3C77C1C97EFE00C8FEC0C8FE80C8FE40C81E0
1C9C579875F1600623EC7906FCD1DBC1E000604CDA68EEE28772424
450 DATA 2424242424410F0C1C9CD0DBD7CFE01C07DFE2CD0112C01
A7ED52CD10BDCD018F21F88ECDAABCC90101002F00000632003AD7913
D32D7915F1600218ACF191100000606360019 10FBC93E00CD0EDC3E
460 DATA 060600CD1A903E0A0602CD1A903E020604CD1A903E040606
CD1A903E0ACD90B821001622439106 0D2108907ECDBE902310F93E0EC
D90BB2106172243912115900605 7ECDBE902310F92103162243912 1
470 DATA D0913E00CD969121 0E1622439121D3913E00CD9691210C17
2243913AD691E6F01F1F1F1FC630CDBE903AD691E60FC630CDBE903E0
2CDDEBB210000114C00CDEABB210000113002CDF6BB210E00113002
480 DATA CDF6BB210E00114C00CDF6BB210000114C00CDF6BB060CC5
3E03CDDEBB115000626 8E5CDEABBE1110C00E5CDF6BB3E0CCDDEBBE11
12E02CDF6BBC17B3D3D4720D6210000CD10BDC9536320202020 2020
490 DATA 2020204E654C6576656CCDDEBB78874F78C6306F16005962
D5C5CDEABBC1D126013E8E906FE5C5CDF6BBC1E116023E7E915FD5C5C
DF6BBC1D1260078C6306FE5C5CDF6BBC1E15459CDF6BB3E01CDDEBB
500 DATA C9F5C5D5E5E579871600625F3EC7906FCD1DBCD10610C5E5
06081AFE00280177132310F6E1CD26BCC110EBE1D1C1F1C9F5C5D5E5E
579871600625F3EC7906FCD1DBCD106080E05E51AFE002801772313
510 DATA 0D20F5E1CD26BC10ECE1D1C1F1C9E5D5C5F5FE30385FFE3A
3058D630171717F52A43917D8787875F16003E1894878787C6076F260
0CD1DBCF1C6CC16975F30011406071AD50E031E00A717CB1317CB13
520 DATA 5797CB4B28023E08CB432802C604772 37A0D20E311FD0719
D11310D721439134F1C1D1E1C92A43913E1FCD5ABB2C7DCD5ABB247CC
D5ABBF1CD5ABB21439134C1D1E1C91416E5D5C53EC7906F26005479
530 DATA 875FCD1DBC7EFE002808FE0328043E80180197C1D1E1C9D5
C53EC7906F26005479875FCD1DBC79EEFFA6F68877C1D1C9D5C5E53EC
7906F26005479875FCD1DBC9777E1C1D1C9E5D5C5F5E5862777300E
540 DATA 233E018627773006233E01862777E1232306037E4FE6F01F
1F1F1FC630CDBE9079E60FC630CDBE902B10E8F1C1D1E1C900000050
100053C00280C4C5028A80A503B4B020F804545030A802F45020180
550 DATA 2A2402108 0302A010180102603038 00E100004801 30D0102
00000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000
560 DATA 000000000000500000000000000000000000000000000000
00000000000000000000000000000034220000000000000000000000
000000000000000000C0C0C0C0C000C0C091333362C0C0C09166CC
570 DATA CC9962C0C066CCCCCCCC99C0C0660C4C8C0C99C0C0668C44
884C99C0C066CC4488CC99C0C066CCCCCCCC99C0C091666C9C9962C0C
0C0919C6C62C0C0C0C066CCCC99C0C0C091CC9966CC62C0C091CC62
580 DATA 91CC62C0C06699C0C06699C0C03333C0C03333C0C0C0C0C0
C0C0C0C00040C0C0C0C08000C0C0903030 60C0C0C0903 03F3F3060C0C
090CC9D6ECC60C0C030CC9D6ECC30C0C035D8B57AE43AC0C035D8A4
590 DATA 58E43AC0C0353F4D8E3F3AC0C0353F4D8E3F3AC0C035D8A4
58E43AC0C035D8B57AE43AC0C030CC9D6ECC30C0C090CC9D6ECC60C0C
090303F3F3060C0C0C0903030 60C0C00040C0C0C0C00000C0C0C0C0
600 DATA C0C0800030303030306000000C0C004C0C0C08000C0C084C0
C0800000C00C0C40C0C0C0C004C04D8E489494C004C04DCF8E4860C04
8C04DCFCF8E68C04884CFCFCFCF0CC0484DCFCFCF0C68C084CFCFCF
610 DATA 0CC060C004CFCF8EC09494C0C00C0C40C0C0C0C0428181C0
C0000000103030342000000C0C0C0C0C0800000C0C0C0C0C0C0C0C0C
0C0F0C0C0F0C0C0C0D03CE0D03CE0C0C0D03CE0D03CE0C0C0843CE0
620 DATA D03C78C0C0843D7A853E78C0C0843F8C4C3F78C0C0843F4C
8C3F78C0C0843F4C8C3F78C0C0843F8C4C3F78C0C0843D7A853E78C0C
0843CE0D03C78C0C0D03CE0D03CE0C0C0D03CE0D03CE0C0C0C0F0C0
630 DATA C0F0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C03C6894
3C943CC0C0296894169416C0C0296894169416C0C0296894169 43CC0C
0296894 16C0C0C0296894169 43CC0C0293C3C169416C0C0290303
640 DATA 169416C0C0293C3C169416C0C0296894169416C0C0296894
169416C0C0296894169416C0C0296894169416C0C03C68943C943CC0C
0C0C0C0C0C0C0C00000C0C0C0C000000 0C03F3F3FC0C00000956AC0
650 DATA 956AC000C0C042C081C0C0C0D1D9E7F3DBE6E2C0F3D9E7F3
DBE6F3C0CCF3E7F3DBF3CCC0F3F3F2F0F1F3F3C0F3F3F3F0F3F3F3C0F
3F3F3F3F3F3C0F3F3F3F3F3F3C0F3F3F3F3F3F3C0CCF3F3F3
660 DATA F3F3CCC0F3D9F3F3F3E6F3C0D1D9F3F3F3E6E2C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C
0C0909090C0C0C0C090C0906060C0C0C0C030C0D090C0C0C0C090C0E4
670 DATA B06090C0C09090D8D890C0C0C0C0D0F0F0E0C0C0C060C064
60C060C0C0C030C06090C0C0C090C090C0C0C0C0C0C090C090C0C0C0C
0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
680 DATA C0C0C0C0C0C06090C0C0C0C0906060 90C090C0C0C0C0C0
608160C0C08190D070C060C0C060C0D0C898C0C0C09090A146E080C0C
0C0F052896460C0C06090 895252C0C0C0906070 46C0C0C0C060D0C8
690 DATA 60E460C0C08190C08042C0C0C0C060C090C060C0C090C060
60C090C0C0C0C0C06090C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C
0C090C0C0C060C0C0C0C0C060C0C0C0C0C0C060C090C0C0C090C0C0
700 DATA C0C0C0C0C0C060C4C060C0C0C0C0C06042E090C0C0C060C4
E042C0C0C060D0C06080C0C0C090C0C846C090C0C0C0C0D0D0C0C0C0C
06090C060C0C0C0C0C0C0C0C060C0C0C0C0C0C0C0C0C0C0C06090
710 DATA C060C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C
0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
720 DATA C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0
C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C0C
0C0C0C0C0C0C0C0C0C0C0C0C08ECC4DC0C058F0A4C0C0D8CFE4C0
730 DATA C0D8CFE4C0C058F0A4C0C08ECC4DC0C0C0C0C0C0C0C0C0C0
C0C0843C78C0C06D9A34C0C06D9A34C0C038659EC0C038659EC0C0843
C78C0C0C0C0C0C0780484848484780010301010101 07C0078840478
740 DATA 8084FC0078840438048478001828488888FC0800FC8480F8
0484780078840 0F8848478 00FC0404001020200078848478848 47800 7
884847C04847 00000000000000000000000000000000000000000
750 DATA 000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000
```

Listing 1.

```
10 REM HEXLOADER - ENTER THE
20 REM STRING OF HEX DIGITS FIRST
30 REM AND THEN THE CHECKSUM
40 PRINT "ENTER START ADDRESS ";:INPUT START
50 PRINT "ENTER FINISH ADDRESS";:INPUT FINISH
60 FOR N = START TO FINISH STEP 8
65 PRINT N;": ";
70 TT=0:INPUT A$:Z=0
80 FOR G=1 TO LEN(A$) STEP 2
90 P=VAL("&H"+MID$(A$,G,2))
100 TT=TT+P:POKE (N+Z),P
110 Z=Z+1:NEXT
120 PRINT " = ";
130 INPUT T$
150 IF VAL(T$)<>TT THEN PRINT "ERROR - ENTER LINE AGAI
N":GOTO 70
160 NEXT
```

SPIX IS NOT a game — it's a real story about power, slavery . . . and chips! On the planet earth, light years away, a worldwide soft war is going on. You, as the last surviving Dragon computer, have to defend the name of your murdered manufacturer against the powerful Softwar Industry. You do this by surrounding parts of the softwar market, thus obliging them to work for you.

The Softwar Industry tries to stop you of course, by electrocuting your circuits, while the involved firms, struggling to avoid colonisation, chase your trail with their evil droids: the Comm, Sing, Orie and Corna computers. These will kill you as soon as they find you. Going backwards while you're surrounding a part causes your death too.

If you can annexe more than 75 percent of the soft war market, then you will receive a bonus chip and move to another country of W-Eurepo, where you will encounter more and stronger competitors.

At 10,000 points, you will receive an extra fuse to survive an electrocution. You can battle with either the keyboard or a joystick as help. Keyboard control uses the four arrow keys and the spacebar. This last key is used to start drawing a line while surrounding a part — the fire button in the case of joystick owners.

A high-score table, and display of score, level, percentage of surface annexed and lives are provided.

You can alter the maximum number of lives with POKE 7348,X

Normally, X=3 but any number between 1 and 15 will do.

To enter the program, type in listing 2, using the hex loader. Type PClear8 before running the hexloader. Save the whole program when finished as

CSAVEM"SPIX",4608,13072,5941

You can type it in in parts too, saving each part as described, but replacing 13072 with the address you reached. Make sure you have a copy of the program before executing it. To play the game, use

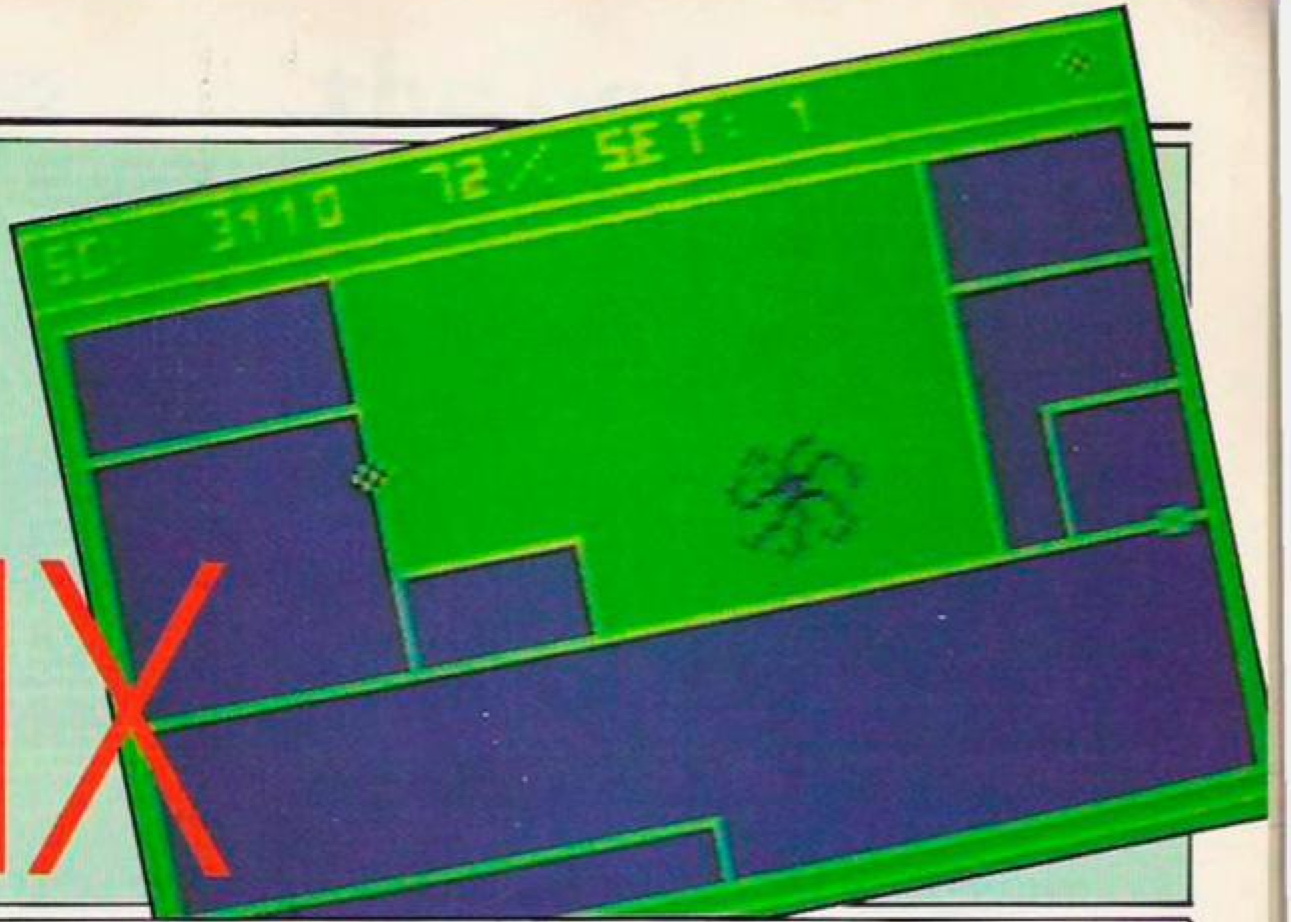
CLOADM"SPIX":EXEC

Listing 2.

```
4608 : 0105454000000000 = 139
4616 : 0015000000150404 = 50
4624 : 0054541500000100 = 190
4632 : 0000000000000001 = 1
4640 : 0104040400000000 = 13
4648 : 0011000000110010 = 50
4656 : 0040400404000500 = 141
4664 : 0000000000000001 = 1
4672 : 0105440000000000 = 74
4680 : 0011000000110040 = 98
4688 : 0054500400000100 = 169
4696 : 0000000000000001 = 1
4704 : 0100440400000000 = 73
4712 : 0011000000110100 = 35
4720 : 0004400404000100 = 77
4728 : 0000000000000001 = 1
4736 : 0105454000000000 = 139
4744 : 0015000000150404 = 50
4752 : 0054540400000100 = 173
4760 : 0000000000000001 = 1
4768 : 1504151511151515 = 147
4776 : 1515000000000000 = 42
4784 : 0000000000000003 = 3
4792 : 0000FC3F30FCFCFC = 1119
4800 : 1114010111101001 = 89
4808 : 11110000A0000030 = 252
4816 : 0A0000A00F00000C = 197
4824 : C0A0303030C0CCC0 = 1004
4832 : 1104150515151501 = 111
4840 : 1515020020C000CC = 472
4848 : 2000002003000030 = 138
4856 : 0A00303C30F0F0FC = 898
4864 : 1104100101011101 = 58
4872 : 1101030002000000 = 53
4880 : 80A00000C000000C = 620
4888 : 2020303030C0CC0C = 624
4896 : 1504151501151501 = 111
4904 : 151500C000300300 = 421
4912 : 82000C00C20000CC = 804
4920 : 2002333030FCCCFC = 1017
4928 : 0AAA000AAA00202A = 570
4936 : 002A0C3003CC0C02 = 451
4944 : A002300A20300302 = 449
4952 : A202000000000000 = 164
4960 : 2555602555A09625 = 687
4968 : 8096300FFC0C0C2A = 659
4976 : A002C02AC0200300 = 623
4984 : A200000000000000 = 162
4992 : 9555602555509625 = 727
5000 : 62563003F00C03C0 = 682
5008 : 0F02C002C00000C0 = 899
5016 : 2A00000000000000 = 42
5024 : 95AA0095A9502025 = 930
5032 : 59560002FEBC0002 = 581
5040 : 00C0300EC00000C2 = 768
5048 : 0A80000000000000 = 314
5056 : 9600009609500009 = 406
5064 : 5550020000000002 = 489
5072 : 20300AA23C00003C = 372
5080 : 220C0030FCFCFC00 = 850
5088 : 95AA009625508202 = 726
5096 : 556000A00C200028 = 553
5104 : 2030002202000000 = 244
5112 : 220300F0CCCCC000 = 877
5120 : 9555009595A0AA00 = 990
5128 : 9580000070300080 = 426
5136 : 000C000000200000 = 428
5144 : 00CC0030FCFCFC00 = 1136
5152 : 255560955A025580 = 672
5160 : 9560000203000200 = 260
5168 : 83302000200000BA = 637
5176 : 00C000300CCC0C00 = 468
5184 : 0AA56255A0025500 = 733
5192 : 255002000C300000 = 203
5200 : 00C00A0220000020 = 396
5208 : 30C00030FCFCFC00 = 1044
5216 : 0009625500025500 = 535
5224 : 955600A0F0000002 = 637
5232 : 0000000000000000 = 128
5240 : 0F00000000000000 = 15
5248 : 2AA5625500095562 = 718
5256 : 5556FF3FCC0CFC03 = 960
5264 : FCC0CFF3FCFF3FCF = 1671
5272 : F3F0C0C30FF0C0C0 = 1509
5280 : 9555625580095562 = 737
5288 : 5656C030CF3CC003 = 874
5296 : 0CC0CC030CC3300C = 678
5304 : 330C33030C303300 = 228
5312 : 9555825600025502 = 667
5320 : 5096CF3FCCCCF003 = 1159
5328 : 0CC0CFC3F0FC3F0F = 1176
5336 : F30C0C030FF00C00 = 537
5344 : 2AAA00A80000AA02 = 552
5352 : A02AC330CC0CC003 = 856
5360 : 0C330C030CC3300C = 345
5368 : 330C0C000C300C00 = 147
5376 : 0000000000000000 = 0
5384 : 0000FF30CC0CFC03 = 774
5392 : FC0C0FF30CC33FCC = 996
5400 : 33F00C030FF00C00 = 573
5408 : 00007E23547E2537 = 463
5416 : 0000151D15054007 = 147
5424 : 400540015001D001 = 424
5432 : 5054745400000000 = 364
5440 : 0000006A6F797374 = 569
5448 : 696368006F720060 = 899
5456 : 6579626F61726400 = 870
5464 : 7B6B7C6A7D000073 = 700
5472 : 7069780006869768 = 881
5480 : 0073636F72650074 = 912
5488 : 61626C650000656E = 615
5496 : 74657200796F7572 = 922
5504 : 00696E697469616C = 874
5512 : 7300000101013030 = 214
5520 : 3030303030010101 = 243
5528 : 3030303030303001 = 337
5536 : 0101303030303030 = 290
5544 : 3001010130303030 = 243
5552 : 3030300101013030 = 243
5560 : 3030303030020100 = 243
5568 : 0202000204000304 = 17
5576 : 0004040002010102 = 14
5584 : 0302020404030404 = 26
5592 : 0404047B637D0074 = 683
5600 : 6F6D006669657273 = 885
5608 : 0062656C6769756D = 869
5616 : 000000000000007E = 126
5624 : 32327E1A357E1900 = 584
5632 : 7E20077E20727E1F = 802
5640 : F457454C434F4D45 = 768
5648 : 20544F204C455645 = 527
5656 : 4C007E205A68224F = 621
5664 : 3254335631354356 = 526
5672 : 3C43563135464556 = 540
5680 : 3C45563135474641 = 523
5688 : 4F3343563C435632 = 546
5696 : 4356313522000000 = 289
5704 : 0000000000000000 = 0
5712 : 0000000000000000 = 0
5720 : 00007E2E6B7E2579 = 563
5728 : 7E2ABA7E30308228 = 746
5736 : FF0C203331313033 = 683
5744 : 292CFF0C28333131 = 669
5752 : 303429292C332C32 = 371
5760 : 00000052007E2DBD = 442
5768 : 0010007401DD0074 = 470
5776 : 001004001D007740 = 232
5784 : 1D00040001000740 = 185
5792 : 1DD0074001000040 = 373
5800 : 01D0077401D00040 = 685
5808 : 00D0224F33543545 = 554
5816 : 4445543444454F32 = 539
5824 : 424F3344434F3254 = 544
5832 : 3241503054344345 = 523
5840 : 4154324250385434 = 537
5848 : 454723424F335432 = 505
5856 : 435031303054344F = 507
5864 : 32454F3345442345 = 490
5872 : 44454F32424F3344 = 530
5880 : 434F325432415030 = 531
5888 : 5434434541543242 = 537
5896 : 50305434454F3343 = 538
5904 : 4F32425431412200 = 427
5912 : DD0012B7FFC5B7FF = 1408
5920 : C2B7FFC0B7FFC9B7 = 1646
5928 : FFC7B6FF2204C78A = 1394
5936 : C0B7FF22398D1718 = 957
5944 : BE0600CC0000ED81 = 718
5952 : 8C120025F9201C0E = 646
5960 : 1202100E06414FEE = 742
5968 : 84EFA4EE02EF2230 = 1096
5976 : 082031A8204C810F = 637
5984 : 25ED398E06008655 = 698
5992 : C620A7898BE0A780 = 1064
6000 : 5A26F70E0620B640 = 753
6008 : C601A784E7881F30 = 944
6016 : 88208C11E025F320 = 861
6024 : 12BE6880108E0600 = 556
6032 : ECA1ED8110BC1200 = 937
6040 : 25F6398DECBE6808 = 977
6048 : 108E1340CC0000EE = 683
6056 : A14CEF81010525F7 = 1023
6064 : 4F31A0163B00165C = 616
6072 : C10E25E88E704D10 = 826
6080 : 8E128ACC0000EEA1 = 949
6088 : 4CEF81010325F74F = 939
6096 : 31A81A30881A5CC1 = 738
6104 : 0F25E88E6E4E108E = 775
6112 : 149C5FEE22EF02EE = 1022
6120 : A4EF8430882031A8 = 968
6128 : 205CC10525EDBD17 = 888
6136 : 478601B775307F75 = 790
6144 : 31B7754306040775 = 854
6152 : 427C7543CC00007F = 705
6160 : 756086108775617C = 884
6168 : 75428D18C8BD1887 = 947
6176 : B67542816C25F0B6 = 1061
6184 : 7543C6203DC3061B = 703
6192 : 1F020E6C41BD174E = 638
6200 : B675438B05B77543 = 877
6208 : 8150223A860DB775 = 746
6216 : 307A7542BD10C8BD = 958
6224 : 1887B67542810422 = 691
6232 : F0B67543C6203DC3 = 1092
6240 : 06011F020E6C41BD = 544
6248 : 174EB675438805B7 = 794
6256 : 7543015022086601 = 570
6264 : B775307E1817BD15 = 731
6272 : F7BD15FD7E15FA7C = 1231
6280 : 7560867560810A25 = 704
6288 : 0F7F75607C7561B6 = 875
6296 : 75618D10398D0139 = 627
6304 : 34100E0190301F26 = 472
6312 : FC351039C63FF7FF = 1141
6320 : 237FFF20B77546C6 = 1017
6328 : 0673FF20B675464A = 851
6336 : 26FD5A26F4863787 = 1035
6344 : FF2339CE7530A6C8 = 1084
6352 : 125F810425058004 = 420
6360 : 5C20F7E7C817A7C8 = 1192
6368 : 18A6C81634426FC8 = 841
6376 : 16A6C813C6203DC3 = 893
6384 : 0600E3C8161F0235 = 541
6392 : 02A7C81610AFC81B = 889
6400 : A6C8100103273101 = 739
6408 : 012741810227338E = 468
6416 : 128234208D174E35 = 623
6424 : 60A6C427375F810A = 706
6432 : 2204313F20023124 = 269
6440 : A6A962B0A7A431A8 = 1109
6448 : 205CC11025F2201C = 672
6456 : 8E1286BD174E3540 = 749
6464 : 20128E12AEBD174E = 674
6472 : 354020008E12AABD = 676
6480 : 174E35407E252027 = 463
6488 : 810A220531A8C020 = 619
6496 : 0431A901E0ECA962 = 950
6504 : 80EDA4ECA962A0ED = 1429
6512 : A820ECA962A2EDA8 = 1270
6520 : 22ECA96282ED2239 = 995
6528 : BD19C88D0316004D = 657
6536 : 8E0401869CC693E7 = 1013
6544 : 8901C0E78901E0A7 = 1090
6552 : 800C041F25F18695 = 864
6560 : C69AE701A7843088 = 1067
6568 : 200C060025F4069E = 751
6576 : B704004A87041F86 = 613
6584 : 980705E08705C086 = 1081
6592 : 970705DF0705FF39 = 1062
6600 : 8E04000600A7000C = 843
6608 : 060025F9398E0425 = 532
6616 : 8F00008E155E8D90 = 917
6624 : E58E05A3BF00088E = 1008
6632 : 15428D90E58E05E6 = 1026
6640 : 8F00008E15DA8D90 = 1041
6648 : E50631B704634CB7 = 957
6656 : 04A34CB704E34CB7 = 916
6664 : 05234CB705635F0E = 640
6672 : 0469100E1508EEA4 = 829
6680 : EF84A622A702EE23 = 1013
6688 : EF0ACE25CF0CEE27 = 1052
6696 : EF0E300840312A5C = 684
6704 : C10525E2398DB8E5 = 1123
6712 : 261C7C756E7C756F = 769
6720 : B6756E810426037F = 710
6728 : 756EB6756F810626 = 810
6736 : E47F756F20DF7F75 = 1082
6744 : 4F014826037C754F = 644
6752 : 8678BD18AC7E1C9A = 947
6760 : 8E0600108E000010 = 322
6768 : AF818C120025F88E = 889
6776 : 0601CC5555ED090B = 766
6784 : E0ED090120ED0901 = 1006
6792 : 80ED098080ED818C = 1147
6800 : 061E25E9A784A789 = 909
6808 : 0120A7890180A789 = 770
6816 : 0800A7890BE08601 = 813
6824 : C6100E07A0A784E7 = 1053
6832 : 013000208C118025 = 539
6840 : F486118E07A0A788 = 1007
6848 : 1F3000208C118025 = 569
6856 : F506010E1160A704 = 934
6864 : A70820A78840A788 = 1005
6872 : 5FA78860A7887FA7 = 1091
6880 : 0900000E0600A784 = 712
6888 : 1F09ED083FA70020 = 939
6896 : ED0900DFED0900FF = 1226
6904 : ED09011FA7090140 = 775
6912 : ED09015FED09017F = 972
6920 : 865187079FB7119F = 923
6928 : 8615071181070701 = 883
6936 : 8E0640108E1200A6 = 554
6944 : A0A7808C06E025F7 = 1109
6952 : BD18EB7F7564865A = 1019
6960 : B775638602B77565 = 936
6968 : 860FB77562398602 = 740
6976 : B77530B775320601 = 833
6984 : B77531B775338D1C = 917
6992 : 7FCA0EF77561BD1C = 1016
```

Tom Fiers defends his dragon to the death in the Soft war to end all Soft wars.

SPIX




```
7000 : 13BD1C56B6754610 = 707
7008 : 2600997E30BDCC08 = 766
7016 : F3FD79B4B6754610 = 1134
7024 : 260089B67560C604 = 772
7032 : 3DF77542F77544B6 = 1105
7040 : 7561B77543B77545 = 950
7048 : CE7530BD18CECE75 = 1113
7056 : 32BD18CE8601B775 = 904
7064 : 50B775547F75747F = 951
7072 : 75597F755B7F755D = 878
7080 : 7F755F8E7551CC19 = 908
7088 : 1EA704E702A704E7 = 964
7096 : 068603B77558B775 = 831
7104 : 5A8664B77576B776 = 1043
7112 : C086EA8776BF7F75 = 1296
7120 : 677F756B7F797D7F = 951
7128 : 797EB604B7755CB7 = 960
7136 : 755E861FB7755287 = 941
7144 : 7556398E0720100E = 599
7152 : 69A0EC81EDA10C12 = 1106
7160 : 0025F7397C7560B6 = 860
7168 : 7560811B1025FF4E = 755
7176 : 7C7561B602B77560 = 870
7184 : 16FF43B6755F3402 = 792
7192 : B6207F755FF67561 = 965
7200 : 3DC30600F3755F1F = 748
7208 : 02350287755F39A6 = 675
7216 : A48D0DA6218D09A6 = 833
7224 : 228D05A6238D0139 = 580
7232 : 01A826118155260D = 619
7240 : 81A4260981902605 = 656
7248 : A601B77546391F21 = 626
7256 : 8DD531A8200DD01F = 983
7264 : 12391F215FA6A48D = 705
7272 : D73123A6A48DD131 = 1028
7280 : 21A6A48DCB31A81C = 952
7288 : 5CC11025E839B601 = 810
7296 : 13270E811B2487B7 = 454
7304 : 75607F7546394420 = 684
7312 : F2860AB775607F75 = 1026
7320 : 4639BD2A714FA700 = 845
7328 : 8C754125F9BD1A68 = 927
7336 : BD1B3EB601B77535 = 766
7344 : 7F7537B6037F798E = 826
7352 : 7F798FB775387F75 = 991
7360 : 3A7F753B7F798106 = 872
7368 : 01B77566BD16068D = 809
7376 : 1603BD171B8D3102 = 501
7384 : 108E14955FEEA4EF = 1063
7392 : 84EE22EF02EE24EF = 1158
7400 : 04A626A706300820 = 597
7408 : 31A0205CC10525E5 = 805
7416 : 34108EC350301F26 = 602
7424 : FC35108664BD18AC = 940
7432 : BD1D5FB67546C604 = 884
7440 : 4A3404B77546BD18 = 713
7448 : ACB67546BD18ACB6 = 1108
7456 : 754635045A26E934 = 657
7464 : 108E2710301F26FC = 582
7472 : 3510810322CFBD1D = 660
7480 : 4EBD1E32BD1F39CE = 830
7488 : 7530BD18CECE7532 = 957
7496 : BD18CE7E16008E07 = 716
7504 : 20108E69A0ECA1ED = 1089
7512 : 818C120025F739BD = 817
7520 : 30F78010B1754625 = 840
7528 : 65B67563B07546C6 = 1060
7536 : 203DFD7560F67562 = 1020
7544 : BD325CC006F17546 = 957
7552 : 2528F075465454C3 = 867
7560 : 0602F375601F0106 = 630
7568 : 55A71F8650A70486 = 930
7576 : 108D1D9E200C5F30 = 579
7584 : 08E0A7845CC10525 = 986
7592 : F639F675625850FB = 1191
7600 : 7546C17022105454 = 726
7608 : 4FF37560C306001F = 767
7616 : 018655A701B605A7 = 694
7624 : 840604BD1D9EB675 = 945
7632 : 63B87546815A2259 = 815
7640 : C6203DFD7560F675 = 1120
7648 : 628D325CC006F175 = 985
7656 : 46251EF075465454 = 732
7664 : C30602F375601F01 = 691
7672 : 8655A71F8650A704 = 930
7680 : 8610300900C0BD1D = 745
7688 : 9EF675625850FB75 = 1163
7696 : 46C170221C54544F = 692
7704 : F37560C306001F01 = 689
7712 : 8655A701B605A704 = 825
7720 : 8604300900C0BD1D = 733
7728 : 9E397F7546108E75 = 804
7736 : 508D0E7C7546B675 = 845
7744 : 463122B1756625F1 = 827
7752 : 39A621C6203DC306 = 748
7760 : 001F01A63F34026F = 426
7768 : 3F1F10E33F1F0135 = 405
7776 : 02A73F4F3000E034 = 771
7784 : 02EC89627FED1FEC = 1104
7792 : 896281ED0135024C = 733
7800 : 308820810525E830 = 667
7808 : 8880A62926037E1E = 668
7816 : 9F7E1E89F1799510 = 1027
7824 : 22068E7C7570391F = 671
7832 : A68962BFA7883FA6 = 1124
7840 : 040403BA1539A704 = 830
7848 : A688200403BA153A = 734
7856 : A788207E31628840 = 888
7864 : 3981032727810127 = 436
7872 : 037E316EA68484C0 = 910
7880 : BA152AA784A68820 = 882
7888 : 84C08A152BA78820 = 909
7896 : A6884004C08A152C = 941
7904 : A7884039EC8484FC = 1176
7912 : C40FBA1533FA1534 = 792
7920 : ED04CC082004FCC4 = 1353
7928 : 0FBA1535FA1536ED = 837
7936 : 8820EC884084FCC4 = 1184
7944 : 0FBA1537FA1538ED = 841
7952 : 88403920EC8840BA = 911
7960 : 1537FA1538ED8840 = 840
7968 : 3981032708A68962 = 639
7976 : 60A788E016FF70A6 = 1178
7984 : 8962E0A7886016FF = 1135
7992 : 66B67563C6203DC3 = 986
8000 : 06007F7561F37561 = 804
8008 : 1F01B67565810222 = 597
8016 : 3CB67564271A8103 = 656
8024 : 271001022706100E = 389
8032 : 16922013108E169C = 555
8040 : 200D108E16A62007 = 430
8048 : BF7569108E16884F = 888
8056 : 7E319DA0E784E601 = 1086
8064 : EAA0E7013088204C = 918
8072 : 810525EC39108E16 = 644
8080 : 8820E4698DDAC655 = 1143
8088 : 81012709E7884210 = 627
8096 : 8E16882007C78840 = 770
8104 : 108E16884F3406EC = 689
8112 : 84AAA0EAA0ED8435 = 1278
8120 : 063088204C810525 = 469
8128 : EC39108E16888103 = 741
8136 : 27131F1083756910 = 522
8144 : 83006026D7CC5555 = 854
8152 : ED88A020CFBF7560 = 1176
8160 : FC75698375601083 = 1013
8168 : 006026C0CC5555ED = 937
8176 : 88602080060F8E04 = 871
8184 : 00A7880C060025F9 = 727
8192 : BD19888E04E78F00 = 918
8200 : 8886008E05E0A780 = 1064
8208 : 8C060025F90E1608 = 604
8216 : BD90E5B675350B70 = 1165
8224 : 8704F88D205A5F0E = 903
8232 : 0400A6A820A7800C = 773
8240 : 05E025F65C108E27 = 801
8248 : 10313F26FCC10F25 = 663
8256 : E639BD324834108E = 808
8264 : 16B2BF00A6A6841C = 883
8272 : FEBDA08D35108F00 = 1065
8280 : A639BE00A634108E = 789
8288 : 161EBF00A6A6841C = 735
8296 : FEBDA08D35108F00 = 1065
8304 : A6398E1608108E06 = 607
8312 : 594FF67538C10427 = 823
8320 : 0AC103270CC10227 = 491
8328 : 13201CEE84EFA420 = 884
8336 : 05CE0000EFA4EE84 = 984
8344 : EF222005CE0000EF = 755
8352 : 22EE84EF242005CE = 922
8360 : 0000EF244C3402B6 = 539
8368 : 01870600350231A0 = 462
8376 : 2030028105250E39 = 500
8384 : A63F34026F3FA621 = 656
8392 : C6203DC30600E33F = 782
8400 : 1F033502A73F397F = 503
8408 : 75707C799A8A799A = 1085
8416 : 81799725007F799A = 896
8424 : BD1E322006BD215C = 621
8432 : BD1E32B675702703 = 722
8440 : 7E165A7C799BB679 = 941
8448 : 9881799825037F79 = 893
8456 : 9BCE7530B6799827 = 1023
8464 : 03BD1522BD18CEBD = 855
8472 : 1525B67535810523 = 579
8480 : 11CE7532B6799827 = 887
8488 : 03BD1522BD18CEBD = 855
8496 : 15257E2A3D757027 = 555
8504 : 037E165A7E165D7C = 606
8512 : 756E7C756FFC756E = 1058
8520 : 810425058601B775 = 610
8528 : 6EC1062701398601 = 541
8536 : B7756F397F754610 = 798
8544 : 8E75507E2517BD21 = 747
8552 : 3F7C75463122B175 = 751
8560 : 6625F039A6A44848 = 910
8568 : AB29B77560F67562 = 1069
8576 : 5858B6756427064A = 694
8584 : B7757220058603B7 = 771
8592 : 7572FB7572F17560 = 1167
8600 : 2400B67560F77560 = 899
8608 : 1F89F07560C10223 = 851
8616 : 0139B67563B77560 = 852
8624 : A621B175602408F6 = 879
8632 : 75608775601F9880 = 960
8640 : 756081022301397C = 561
8648 : 757039BD20C0E628 = 969
8656 : C102102200D7A629 = 662
8664 : 2719C10227106C29 = 463
8672 : 4C81041026FF8D6F = 770
8680 : 296CA416FF866A29 = 871
8688 : 16FF81CC0000FD75 = 900
8696 : 6B87756DE620A6C9 = 1153
8704 : 6200043001102603 = 592
8712 : 7C756BA6C962C004 = 1137
8720 : 30811026037C756C = 583
8728 : C102270FA6C962A0 = 874
8736 : 840F810526127C75 = 578
8744 : 6D200DA6C962A004 = 911
8752 : C0814026037C756D = 776
8760 : B6756D2720B6756C = 886
8768 : 8B756B8102275281 = 792
8776 : 012735C10227056C = 440
8784 : 2916FF200603A729 = 695
8792 : 6AA416FF17B6756C = 977
8800 : 270CB6756B271086 = 694
8808 : 756E810226096A21 = 544
8816 : 8603A72816FEFD6C = 981
8824 : 218604A72816FEF4 = 898
8832 : B6756C270A86756E = 865
8840 : 810226E816FF8CB6 = 1051
8848 : 756E810226D816FF = 889
8856 : B2B6756E81021027 = 773
8864 : FFA98101270120C6 = 1032
8872 : 7F756B7F756C7F75 = 947
8880 : 6DC1042706A6C962 = 816
8888 : 882004A6C962C004 = 953
8896 : 30811026037C756D = 504
8904 : A6C962A0840F8105 = 906
8912 : 26037C756CA6C962 = 855
8920 : A804C0814026037C = 842
8928 : 7560B6756D271CB6 = 881
8936 : 756C8B756B810227 = 886
8944 : 5381012735C10327 = 540
8952 : 856C2116FE766A21 = 679
8960 : 16FE71B6756C270C = 847
8968 : B6756B2714B6756E = 874
8976 : 8102260D8602A728 = 525
8984 : 8603A7296AA416FE = 891
8992 : 538601A7286C2916 = 596
9000 : FE4AB6756C270AB6 = 966
9008 : 756E810226C816FF = 988
9016 : BCB6756E81021027 = 783
9024 : FFB320D0B6756E81 = 1212
9032 : 021027FFA8810127 = 649
9040 : D016FFC0AEC8184F = 1157
9048 : 10AE89628010AF84 = 876
9056 : 10AE89628210AF02 = 748
9064 : 4C308820811025E8 = 786
9072 : 7F7992A6442714BD = 876
9080 : 213FB60113811E22 = 491
9088 : 0AA64727718D24B9 = 889
9096 : 16009A8D24418675 = 765
9104 : 602663A6C42600A6 = 887
9112 : 41275B810A2757A6 = 626
9120 : C4271F810A250D80 = 583
9128 : 0A877561A6C81200 = 967
9136 : 75612003A8C81281 = 767
9144 : 68220701042303A7 = 483
9152 : C812A641272F810A = 674
9160 : 2582800A81022304 = 347
9168 : 6A4120EEA641810A = 811
9176 : 258D800A877561A6 = 751
9184 : CB13B075612003AB = 815
9192 : CB1381482207810D = 603
9200 : 2303A7CB1339B675 = 780
9208 : 6F8104240F4A2706 = 414
9216 : 8B0AA7C4200A8B0B = 784
9224 : A7C420040003A7C4 = 893
9232 : B6756E8875308103 = 893
9240 : 22074A8B0AA74120 = 528
9248 : 048003A7418D2441 = 657
9256 : B675601027FF707C = 941
9264 : 7992B67992810525 = 887
9272 : 037E24B2BD213F20 = 660
9280 : B57F7560AEC818A6 = 1088
9288 : C42732810A2220A8 = 661
9296 : C818810425023001 = 445
9304 : 30055FA689628026 = 715
9312 : 0A3088205CC11425 = 568
9320 : F220127C756039A8 = 857
9328 : C818810E2502301F = 485
9336 : 301F5F208EA64126 = 697
9344 : 0139810A221AC620 = 487
9352 : 3DE3C8181F013089 = 732
9360 : 0200BD0530028D01 = 340
9368 : 39EC89620026CC39 = 955
9376 : 000AC6203DFD7568 = 906
9384 : ECC81BD375601F01 = 890
```

(listing 2 continued on next page)

(listing 2 continued from previous page)

```
9392 : 20E06FC46F417E23 = 900
9400 : 9FA6C817E6C01010 = 1018
9408 : 27FF33D175622513 = 793
9416 : B075622720010F22 = 640
9424 : 04860B2002B60CA7 = 496
9432 : C42012B67562A0C8 = 1003
9440 : 17810F2204860120 = 372
9448 : EE860220EAA6C813 = 1025
9456 : B175632510807563 = 838
9464 : 810F2204860B2002 = 361
9472 : B60CA74139B67563 = 833
9480 : A0C013810F220486 = 695
9488 : 0120EF860220EB8D = 864
9496 : 21C8BD213F7C7546 = 832
9504 : B675463122B17566 = 840
9512 : 25ED3910AEC81BA6 = 914
9520 : 412601397EC195A6 = 566
9528 : C8178779948B030E = 959
9536 : 7576877572E680C1 = 1200
9544 : 64260139F1757223 = 703
9552 : 07E680C16426F539 = 998
9560 : F17994240220F2A6 = 988
9568 : C013B77995E60901 = 1040
9576 : 49F1799525E38B8D = 1000
9584 : B779957E1E8C7570 = 978
9592 : 397F75737C799CB6 = 999
9600 : 799C81799925067F = 898
9608 : 799C7E29837F7573 = 934
9616 : B6756410270087F6 = 883
9624 : 7565C10227094C81 = 666
9632 : 041025000E20004A = 313
9640 : 102600077A75627F = 653
9648 : 7564B675742764FC = 1023
9656 : 7567C375761F01B6 = 864
9664 : 75624CA780C664E7 = 1115
9672 : 84FC797DC378501F = 1056
9680 : 02B67565A7A48605 = 872
9688 : A721FC797DC30001 = 894
9696 : FD797DFC7567C300 = 1166
9704 : 01FD7567C664B675 = 1071
9712 : 65B1032400B67563 = 680
9720 : A7890149E789014A = 821
9728 : 201601032707B675 = 531
9736 : 6380032003B67563 = 663
9744 : A7890149E789014A = 821
9752 : BD29C7B675658102 = 960
9760 : 1027016F01031027 = 354
9768 : 020C8104102702A4 = 368
9776 : 16009AB77564B675 = 875
9784 : 658101102700F681 = 661
9792 : 021027019B810310 = 361
9800 : 27020116031D7F75 = 468
9808 : 73B6754F2739B67F = 858
9816 : 8D2A260586018775 = 661
9824 : 7386BF0D1F102700 = 667
9832 : 5686DF0D17102701 = 663
9840 : 1D86F7B0DF102701 = 622
9848 : 8A86EF8D07102702 = 764
9856 : 507E2983B7FF02F6 = 1064
9864 : FF00C47FC15F39BD = 1112
9872 : 8012B6FF80817F27 = 878
9880 : 0781FF27037C7573 = 789
9888 : B6015A813A221801 = 647
9896 : 05102500E1B6015B = 557
9904 : 813A1022021D8105 = 400
9912 : 102501777E29837C = 595
9920 : 7564F67562F77546 = 1112
9928 : 7C75622006B67562 = 774
9936 : B77546BD2970B675 = 1011
9944 : 62811E231AF67546 = 751
9952 : F775627F7564B675 = 1105
9960 : 742709BD16607F75 = 715
9968 : 747F75737E2983BE = 963
9976 : 7560B67564010127 = 701
9984 : 023001B67574275E = 599
9992 : A6004104F0015027 = 987
10000 : 11A6B0EC8110263F = 789
10008 : A68900A081102734 = 699
10016 : 2035A68962C0043C = 870
10024 : 01142628BD16607F = 661
10032 : 75747F7573B60187 = 910
10040 : 7565BD2970BE7560 = 963
10048 : 4F10AE09620010AF = 823
10056 : 844C300820010525 = 595
10064 : F07E29837E165AA6 = 942
10072 : 8962C00430011026 = 790
10080 : D480166020CFA689 = 1061
10088 : 62C0043F811527C5 = 871
10096 : 0601877574FE7562 = 1020
10104 : FF7906B675732685 = 1143
10112 : F67546F775627F75 = 1139
10120 : 647F757420A70603 = 796
10128 : B77564BD2970B675 = 1041
10136 : 62010022117F7564 = 622
10144 : B6757427068D1660 = 767
10152 : 7F75747E27E08E75 = 1056
10160 : 60B67574275DA680 = 943
10168 : 40043F81152711A6 = 631
10176 : 08E10110263CA689 = 907
10184 : 00A1011027312032 = 476
10192 : A60962C104F00150 = 1175
10200 : 26250016607F7574 = 742
10208 : 8602877565BD2970 = 879
10216 : BE75604F10AE0962 = 907
10224 : 0010AF844C300820 = 743
10232 : 010525F07E29837E = 835
10240 : 165AA68962C18430 = 886
10248 : 811026D48D166020 = 734
10256 : CFA68962C104F001 = 1302
10264 : 5027C58601877574 = 867
10272 : FE7562FF7906B675 = 1270
10280 : 7326B57F75647F75 = 922
10288 : 7420AD7C75647A75 = 901
10296 : 63BD2970B6756381 = 968
10304 : 0922117F75647C75 = 645
10312 : 63B675742703BD16 = 767
10320 : 607E2983BE7560B6 = 979
10328 : 7574274BA6B8E184 = 1006
10336 : 308110260DA68962 = 645
10344 : 618430811027037E = 590
10352 : 165AA68962C18430 = 886
10360 : 81102606BD16607F = 623
10368 : 757486603B7565BD = 960
10376 : 2970BE7560EC8962 = 1027
10384 : E0ED8860EC096300 = 1165
10392 : ED0900A0EC096320 = 1006
10400 : ED0900A07E2983BD = 1021
10408 : 3281128430811027 = 561
10416 : D18601B77574FE75 = 1131
10424 : 62FF7906B6757326 = 1060
10432 : C17F75647C75637F = 1004
10440 : 757420B67A756320 = 817
10448 : 817C75647C7563BD = 1047
10456 : 2970B67563815825 = 808
10464 : 147F75647A7563B6 = 884
10472 : 75742706BD16607F = 712
10480 : 75747E2983BE7560 = 934
10488 : B675742749A60900 = 830
10496 : A184308110260DA6 = 703
10504 : 8963218430811027 = 633
10512 : 037E165AA68962C1 = 835
10520 : 843081102606BD16 = 580
10528 : 607F757486048775 = 894
10536 : 65BD2970BE7560EC = 1082
10544 : 8962A0ED0820EC89 = 1173
10552 : 6200ED04EC096260 = 1162
10560 : ED08E07E2983A689 = 1198
10568 : 63018430811027D4 = 676
10576 : 8601B77574FE7562 = 1020
10584 : FF7906B6757326C4 = 1158
10592 : 7F75647A75637F75 = 926
10600 : 7420B97C75632084 = 885
10608 : 7F7561B67563C620 = 969
10616 : 3DF37561C30600FD = 972
10624 : 756039BD31248675 = 843
10632 : 3AC60A3DF8753BC1 = 947
10640 : 4B2231B6790E2610 = 657
10648 : B6753D810126097C = 661
10656 : 75387C790EB02072 = 895
10664 : B67574270DB67562 = 864
10672 : 4848BB7563BD18AC = 932
10680 : 2007BE799D301F26 = 624
10688 : FC7E16007E166310 = 663
10696 : 0E70500E7576A680 = 1013
10704 : BD2A27A6A0010324 = 764
10712 : 2C8102271CECC83F = 741
10720 : BD32DC12EDC83FA6 = 1143
10728 : C83F81012632A61F = 677
10736 : 0105252C6FC03E20 = 620
10744 : 27ECC0408D32DC12 = 1016
10752 : EDC840201BA6C840 = 990
10760 : 7E32E5C840A6C900 = 1036
10768 : A00A10A7C900A006 = 976
10776 : 10A7C860A7C90000 = 975
10784 : A6A4010526A03987 = 910
10792 : 79037F7902A60901 = 934
10800 : 49C6203DC30600F3 = 808
10808 : 79021F0339B67570 = 753
10816 : 27037E165A7C799C = 681
10824 : B6757427087C799C = 863
10832 : 7C799C8D14B6799C = 1021
10840 : 81799925097F799C = 901
10848 : 8D1F397E16007E16 = 573
10856 : 5D8E799D301F26FC = 930
10864 : 39061CD779970603 = 813
10872 : B779908614877999 = 1067
10880 : 8E04558F799D8E75 = 959
10888 : 3C390D068D32047E = 841
10896 : 2879863FB7FF237F = 961
10904 : FF20869673FF201F = 1004
10912 : 895A26FD4A26F506 = 1009
10920 : 3787FF2339B67570 = 996
10928 : 0103241DB6757601 = 791
10936 : 75472200B6756201 = 807
10944 : 754725297E284FB6 = 696
10952 : 7562817547221E20 = 676
10960 : 7EB676C001754322 = 1013
10968 : 0AB67563B1754325 = 806
10976 : 0C206CB67563B175 = 844
10984 : 43220220628E7578 = 612
10992 : A61F8D2A271F325F = 643
11000 : B678510102222931 = 630
11008 : A0E05CA6A427F8F7 = 1348
11016 : 75601F325F31A020 = 630
11024 : 5CA6A427F8F17560 = 1163
11032 : 2207860387756C20 = 610
11040 : 7D860487756C2076 = 821
11048 : 31215CA6A427F9F7 = 1039
11056 : 75601F325F313F5C = 593
11064 : A6A427F9F1756022 = 1106
11072 : 07860187756C2056 = 668
11080 : 860287756C204FB6 = 837
11088 : 757081032410B676 = 713
11096 : C081754325040603 = 731
11104 : 20128604200E8675 = 533
11112 : 7681754725048602 = 660
11120 : 20028601B7756C20 = 609
11128 : 25FC797DC3784F1F = 960
11136 : 01A6841F03B67565 = 733
11144 : B77570A6820C7050 = 1040
11152 : 1023FF1981757010 = 753
11160 : 27FFF087756CF675 = 1305
11168 : 625050FB75645CB6 = 1016
11176 : 75642602CB04F775 = 828
11184 : 72F67570C1022713 = 842
11192 : C103271AC1042721 = 530
11200 : B675724A4A488779 = 937
11208 : 7F201DB675724C4C = 753
11216 : 4887797F2012B675 = 852
11224 : 638B0348B7790020 = 777
- 11232 : 07B6756348B77900 = 909
11240 : F67546C1C7270BC6 = 1073
11248 : C7F77546F6756C16 = 1126
11256 : FFBABD1D4EBD29C7 = 1166
11264 : 2005B7312C3100BE = 552
11272 : 00A634100E2C03BF = 614
11280 : 00A6B600B71CFEBD = 1002
11288 : A9AF35100F00A6BE = 960
11296 : 00A634100E1667BF = 692
11304 : 00A6A6041CFEBDAC = 1107
11312 : 0735100F00A6BE75 = 868
11320 : 4BA61F81AA2633A6 = 826
11328 : 0401AA262DA60BE1 = 913
11336 : 81AA26267E32C181 = 873
11344 : 0127148102270C81 = 371
11352 : 0327048603200A86 = 359
11360 : 0420068601200286 = 345
11368 : 0287756C7F754616 = 746
11376 : FF2C8D1BCD0605D7 = 1072
11384 : 78508664B77576B7 = 1035
11392 : 76C07F797D7F797E = 1057
11400 : 7F75677F75688E07 = 844
11408 : A07F79027F7983A6 = 1083
11416 : 002609FC7982C300 = 873
11424 : 01FD79828C110025 = 827
11432 : EECC00F37F757003 = 1100
11440 : 00177C7570100379 = 692
11448 : 0222F4FC79848379 = 1213
11456 : 828D3292FC7982FD = 1271
11464 : 79842000B675707F = 823
11472 : 753A7F7538810A25 = 654
11480 : 07800A7C753A20F5 = 721
11488 : B7753BB6753B8E06 = 865
11496 : 4D8D11B6753A2705 = 636
11504 : 8E064C8D077F7574 = 732
11512 : 7F757339108E12A0 = 752
11520 : 5F3404E6A6E70435 = 963
11528 : 045C30882031A820 = 561
11536 : C10525ED39BD1F39 = 806
11544 : 860EF67565C10223 = 842
11552 : 435F8D14860FC601 = 671
11560 : 8D0E8610C6028D08 = 654
11568 : 8611C6038D02202C = 571
11576 : F77561F67565C104 = 1122
11584 : 270FC604F07561F7 = 957
11592 : 7561C10425037F75 = 695
11600 : 61F67561B1756327 = 989
11608 : 078004815A25F539 = 708
11616 : F7756439B6757426 = 974
11624 : 32B67564272DB675 = 832
11632 : 6581022326B67564 = 704
11640 : D179A027058779A0 = 966
11648 : 2019810326157F79 = 496
11656 : A0B675657F756481 = 1033
11664 : 0427057A75632003 = 421
11672 : 7C75630E7530A680 = 941
11680 : 27068C753425F739 = 695
11688 : 862B8B7564A68681 = 1007
11696 : 02240220044A4A8B = 363
11704 : 0AA71F20E1108303 = 615
11712 : E825008303E07C75 = 884
11720 : 3E20F21083006425 = 620
11728 : 080300647C753F20 = 575
11736 : F2C10A2507C00A7C = 815
11744 : 754020F5F87541C1 = 1084
11752 : 0A250AC00A7C7540 = 564
11760 : F0754120EFF77541 = 1122
11768 : F67540C10A2505C0 = 864
11776 : 0A7C753F8D3266F6 = 901
11784 : 753FC10A2505C00A = 627
11792 : 7C753EF7753FF675 = 1093
11800 : 3EC10A2505C00A7C = 633
11808 : 753DF7753EF6753D = 1028
11816 : C10A2505C00A7C75 = 688
11824 : 3CF7753D8E064410 = 717
11832 : 8E753CA6A4261A10 = 729
11840 : BF756034108E753C = 791
11848 : A68026098C756025 = 779
11856 : F73510200DA6A435 = 742
11864 : 1034308D2CFC3530 = 782
11872 : 30013121108C7542 = 470
11880 : 25D1397F757207FF = 1099
11888 : C5B7FFC207FFC0B7 = 1642
11896 : FFC9B7FFC7B6FF22 = 1564
11904 : 84CF8AC8B7FF2220 = 1181
11912 : 1B863FB7FF237FFF = 1079
11920 : 20C696061473FF20 = 936
11928 : 4A26FD5A26F50637 = 927
11936 : 87FF23398DE38D17 = 1110
11944 : 100DDE7C75728675 = 1041
11952 : 72810F25897A7538 = 775
11960 : B675302772B67576 = 925
11968 : 01642757B6785081 = 866
11976 : 0127178102271981 = 387
11984 : 0327057A76C02012 = 529
11992 : B676C00803B776C0 = 1127
12000 : 20087A75767A7576 = 754
12008 : 2000B67576B77562 = 847
12016 : B676C00B02C6203D = 924
12024 : C306007F7561F375 = 902
12032 : 611F01A689620126 = 697
12040 : 037A7562B676C087 = 1015
12048 : 7563810A2405860A = 540
12056 : B77563BD183E8D1D = 895
12064 : 4E7F75647F757486 = 916
12072 : 05B770507E1CCF0E = 891
12080 : 080ACC0000A70900 = 657
12088 : C0A7005CC10C25F5 = 1066
12096 : 5FA704A714300820 = 797
12104 : 5CC10725F48E08AB = 897
12112 : 108E140A7F75465F = 725
12120 : A6A0A7005CC10825 = 954
12128 : F730081531A8157C = 814
12136 : 7546B67546010525 = 727
12144 : E6BDBBE527FB0E75 = 1384
12152 : 3C100E7900A6080D = 900
12160 : 30A7A00C754225F5 = 980
12168 : 8E150EEC8410B379 = 909
12176 : 8025261083790826 = 701
12184 : 16EC0210B3790A25 = 751
12192 : 1010D3790A2600EC = 760
12200 : 041003790C250A30 = 555
12208 : 0A8C150623D57E17 = 750
12216 : 353410BD19008E04 = 617
12224 : E4BF00000E05C006 = 1060
12232 : 80A7008C060025F9 = 855
12240 : 8E15758D90E53510 = 911
12248 : BF7560BDBBE527FB = 1299
12256 : 0040B704F9BDBBE5 = 1233
12264 : 27FB00400704FABD = 1100
12272 : BBE527FB0040BD32 = 1137
12280 : 758E7560301D8F75 = 905
12288 : 60100E15B3A6A4A7 = 951
12296 : 2A313F100C756024 = 607
12304 : F48E7560FC04F9ED = 1309
12312 : 848604FBA702FC79 = 1111
12320 : 00CD03FC790AED05 = 1129
12328 : FC790CED077E1735 = 959
12336 : 7C75350E9C403D30 = 765
12344 : 1F26F8BD1606BD1A = 752
12352 : 687F7546CC03E8BD = 1046
12360 : 1685B67535010A24 = 682
12368 : 080E06560D2CFC20 = 759
12376 : 07800A7C754620ED = 725
12384 : B6754627060E0655 = 647
12392 : 8D2CFC8D1B3E8E15 = 926
12400 : DA0603F67535C10A = 942
12408 : 2502C60A3DA605B7 = 790
12416 : 75667F753A7F753B = 824
12424 : 5C2023A6056F246F = 716
12432 : 2781022714810426 = 400
12440 : 060601A724200A81 = 515
12448 : 01270A8601A72420 = 420
12456 : 048601A72739108E = 560
12464 : 75300DD7100E7532 = 846
12472 : 0DD17E1CCFA6A01F = 1076
12480 : 2620A6A023261BA6 = 670
12488 : A9013F2615A6A901 = 628
12496 : 43260FA6A9023F26 = 558
12504 : 09A6A9024326037E = 580
12512 : 18667C7560867560 = 861
12520 : 811825087C7561B6 = 673
12528 : 028775607E1B56B6 = 819
12536 : 7563011225013986 = 592
12544 : 12390E0BABCC0000 = 603
12552 : A70900C0A7005CC1 = 1076
12560 : 0825F55FA710A784 = 875
12568 : 3088205CC10725F4 = 789
12576 : 8E08CC398D2D15B6 = 851
12584 : 7574270139B67565 = 730
12592 : 81012701397F7561 = 568
12600 : B675630802C6203D = 830
12608 : C30600F375611F01 = 690
12616 : B6756281032312A6 = 748
12624 : 1F810126026F1FA6 = 509
12632 : 081F810126036F08 = 585
12640 : 1F39A680400403BA = 775
12648 : 153BA7804039EC84 = 872
12656 : 84F0C43F8A152DFA = 1133
12664 : 152EED84EC802004 = 972
12672 : F0C43F8A152FFA15 = 1024
12680 : 30ED0020EC804004 = 1021
12688 : F0C43F8A1531FA15 = 1026
12696 : 32ED804039E6A4F6 = 1184
12704 : 7565C1012632F675 = 863
12712 : 74262DF67562C103 = 856
12720 : 2326E6803FC40FC1 = 906
12728 : 09260AE6803FC4FE = 936
12736 : CA02E7803FE6805F = 1095
12744 : C40FC100260AE600 = 826
12752 : 5FC4FECA02E7005F = 1211
12760 : 8D0F30013121E6A4 = 681
12768 : 8D07301F31217E1F = 466
12776 : 84E6A4C1002741E6 = 1053
12784 : A4C4C0C1002706E6 = 1020
12792 : 84C43FE704E6A4C4 = 1344
12800 : 30C1002706E604C4 = 844
12808 : CFE704E6A4C40CC1 = 1365
12816 : 002706E604C4F3E7 = 1077
12824 : 04E6A4C403C10027 = 957
12832 : 0DE604C4FCE704E6 = 1416
12840 : 84EAA4E7043920F7 = 1229
12848 : 20F5B679A9816327 = 1016
12856 : 0B6663B779A97F75 = 961
12864 : 407E20427F754039 = 653
12872 : 0E1200100E7B0CEC = 689
12880 : 81EDA10C122025F7 = 1001
12888 : 8E00A6394FC10122 = 720
12896 : 02C602585039C10A = 638
12904 : 2507C00A7C753F20 = 582
12912 : F5F7754039B704FD = 1168
12920 : 8EC3503D301F26FB = 846
12928 : 39B67563810D2505 = 639
12936 : A689628139A68962 = 908
12944 : 613934068E780C10 = 585
12952 : 8E0600EC01EDA90C = 931
12960 : 00EDA84031220C78 = 815
12968 : 2C25F035068D1685 = 724
12976 : 8D2072B67535108E = 845
12984 : 75358E06568D2CFC = 889
12992 : 397C79DFB679DF81 = 1180
13000 : 022406B6756C7E2C = 621
13008 : 4F7E2C727F75467F = 804
13016 : 79DF39A404C08A15 = 1048
13024 : C40FCA503984CF8A = 1027
13032 : 10A7C840A6C900A0 = 974
13040 : 84CF8A10A7C900A0 = 1021
13048 : A6C86084CF8A10A7 = 1122
13056 : CB60A7C900807E2A = 960
13064 : 2066666664666466 - 742
13072 : 66FFFFFFFFFFFF = 1632
```

the only choice

Kuma AMSTRAD CPC464 software



Holdfast | Gems of Stradus | Star Avengers | Galaxia | Music Composer



Logo | Database | ZEN Assembler | EASIVAT | Home Budget

An outstanding selection from Kuma's rapidly expanding range of Entertainment and Application Software for the Amstrad CPC 464 Micro-computer.



BOOK ● *The Amstrad CPC 464 Explored.*

This superb book is designed to let every CPC 464 user, at whatever level, get the most from his computer. After an introductory section on the special Basic features, the book looks in depth at the excellent sound and graphic facilities.

Now available from selected branches of Co-op, Granada, LASKYS and John Menzies

Kuma Computers Ltd., Unit 12, Horseshoe Park, Horseshoe Road, Pangbourne, Berks RG8 7JW.

Please send full catalogue on Amstrad CPC464 products.

Name ....

Address ....

....

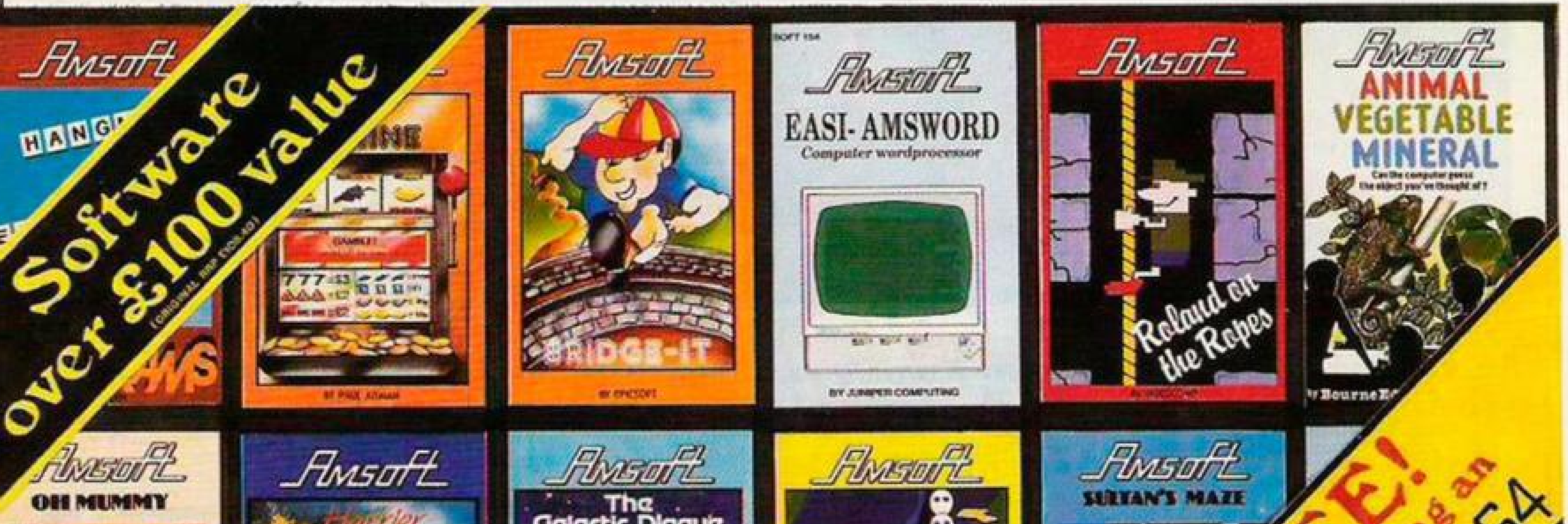
.... Phone..............................

I own an Amstrad CPC 464 computer ☐

Trade Enquiries Phone 07357-4335

Visitors wishing to call at our Pangbourne Manufacturing and Distribution Centre are advised to phone 07357-4335 first for an early appointment.

Now Amstrad gives you over £100 to play with.



£239
complete with green screen VDU (GT64)

£349
complete with colour monitor (CTM640)



CPC464 complete with monitor and datacorder

Join in the fun.

Our CPC 464 User Club is a must for all owners.

As a member, you'll be the first to know about the additions to the range. Hardware and software.

We'll provide you with your own privilege card and send you free, a year's subscription to the Amstrad User magazine packed with programs, features and information.

You can enter competitions for valuable prizes and keep up with all the latest Amstrad developments.

And with the free software pack, we think that's pretty good for starters.



Free software worth over £100 is the name of the game.

Along with your new CPC 464 (colour monitor or green screen VDU) you'll receive a software starter pack of 12 (yes, twelve) cassettes absolutely free.

There's something for everyone. From Roland-on-the-Ropes to Easi-Amsword word processing.

That's *one* good reason for making Amstrad's CPC 464 your new home computer.



CPC 464 green screen VDU (GT64)

Here are the others.

The CPC 464 comes complete with its own colour monitor or green screen VDU and built-in cassette data recorder. (Optional disc drives are now available with CP/M* and DR. LOGO* supplied as standard.)

It has a typewriter-style keyboard and numeric keypad. 64K of RAM, 32K of ROM and a very fast extended BASIC with real time features.

And it costs a great deal less than the price of a comparable system.

The CPC 464 can handle the trickiest computer games (if you can), revision for exams, cataloguing and budgeting.

And it comes complete and ready-to-go.

Amsoft. A ton of software.

Yes. Now there are over 100 titles from Amsoft and independent publishers. And the list is still growing.

Arcade games, educational programs and business applications – they're all designed to make the most of the CPC 464's impressive graphics, stereo sound and processing abilities.

And with Amstrad's *speedloading* capability, even complex programs can be loaded quickly. The optional disc system will load a game in around 6 seconds.

When you feel the need to upgrade your Amstrad system to disc drive, just send your favourite Amsoft cassettes to us and we'll send you back the same Amsoft programs on Amsoft disc for just £4.95 per cassette, the price of a blank disc.



AMSTRAD
ONE GREAT IDEA AFTER ANOTHER

BOOTS COMET Currys Dixons Greens John Menzies RUMBELOWS spectrum WHSMITH WIGFALLS WOOLWORTH AND OTHER COMPUTER STORES

*CP/M and DR LOGO are Trade Marks of Digital Research

I'd like to know more about the exciting CPC 464 complete computer system. Please send me literature right away.

NAME

ADDRESS

POSTCODE

To: Amstrad Consumer Electronics PLC, Brentwood House, 169 King's Road, Brentwood, Essex CM14 4EF.

PRICES CORRECT AT TIME OF GOING TO PRESS. NOT ALL STORES STOCK ALL PRODUCTS. YC8



THE FUTURE OF MANKIND IS IN YOUR HANDS!
DROPZONE
It's the year 2085. Only a handful of people have survived the robot wars that rocked the Solar System. In a final desperate bid for survival a Tacheon propelled star cruiser has been developed on Earth to transport survivors to new star system. But the cruiser relies on rare Ionian crystals for its power; crystals which are only abundant on Jupiter's second moon Io. A moonbase established on Io is under constant attack by aliens from Jupiter. It is your mission to escort the men and their precious crystals safely from the surface of Io to the Dropzone where the landing pad is located. You are equipped with the latest pulse-laser back-pack system which will make you invisible and indestructible for short periods of time. You must succeed, the future of the human race depends on you alone!
64
ATARI
Super smooth scrolling screen
High resolution graphics
Nine different alien attackers
Scrolling high speed scanning viewer
Exploding volcanoes
Ultra fast action
Cassette £9.95
Disk £14.95
U.S. GOLD
AI American Software
U.S. GOLD Ltd., Unit 10, The Parkway Industrial Centre, Heneage Street, Birmingham B7 4LY.
Telephone: 021-359 3020. Telex: 337268.

Memory Map (in Hex)

Start	Finish	Comments
0800	0FFF	Character set 1.
1000	17FF	Characters for planet animation.
1800	10FF	Characters for landscape.
1E00	23FF	Colours for landscape.
2400	37FF	Sprite data.
3800	3FFF	Character set 2.
4000	7BC8	Main machine code with various data in between, this includes data for music and different sheets and text.

Entry points to important routines

Address (Hex)	Purpose of Routine/ Description
410D	Start of interrupt routine.
4256	Start of music operating system.
43A9	Animation of satellites.
443A	Animation of planet characters.
4580	Clear screen downwards.
45EC	Clear screen upwards.
476B	Scroll planet left.
481A	Scroll planet right.
4909	Start of a series of sprite manipulation routines.
4AF0	Start of first title screen.
5352	Start of second title screen.
5307	Set up registers for a new game.
5590	Plot landscape on screen.
559E	Explode ship on to the screen.
56FE	Handle joystick control.
5838	Set up registers for another go, after losing a life.
592F	Plot satellites on screen.
5989	Clear satellites on screen.
5AA1	Handle the bullets from your ship.
5B6D	Handle junkies, troopers and stormers.
5FE6	Put sprites on screen if within range.
6032	Plot multicoloured block on radar.
60D7	Clear radar.
6770	Begin game.



SPACE JUNK

Nalin Sharma continues his excellent shoot'em up for the CBM-64.

SPACE JUNK WAS written with an assembler and it took about three months. Also I used a variety of graphics programs to produce the characters, sprites and animation.

The actual code for the gameplay is something extra special because it is almost entirely raster-interrupt driven. This accounts for the ultra-smooth graphics. Why? Because the interval between graphic movements is always equal and this makes the graphics very slick indeed. The sound effects during play and the music soundtrack are also interrupt-driven which make them far more effective.

A disassembly of the code would obviously be beyond the scope of this magazine, so instead I will give a basic breakdown of the program consisting of a rough memory map and the addresses of various important routines such as the scrolling — see memory map.

I very much hope that the memory map together with the entry points to important routines will prove both interesting and useful. I also hope that it gives you an insight into the depth and complexity of the task of writing a machine-code game of this length.

Listing 3.

```
1940: 1C1F23261C1F2326 108
1948: 1C1F23261C1F2326 108
1950: 1C1FD4207D797A7A 319
1958: 797E20D7CBCED1C5 51D
1960: C8CBCED1C5C8CBCE 658
1968: D1C5C8CBCED1C5C8 655
1970: CBCED1C5D4202020 463
1978: E145524F4C40434F 2E5
1980: 4D505554455253E3 313
1988: 20E8E92020202020 291
1990: 208B8E9120202020 24A
1998: 20B3B6B8BB202020 35C
19A0: 207FE47F20202020 282
19A8: 9AF5F5F5F5F5F5F5 74D
19B0: F5F5F5F5F59C2020 5A5
19B8: 202020202094E620 23A
19C0: 202020E520202020 1C5
19C8: 9AEFEFEFEF9C2020 532
19D0: 20202020202020E6 1C6
19D8: 942020202020BDE7 2D8
19E0: C0C220203220207D 2B1
19E8: ED7E20202054454E 2B2
19F0: 4E49532020202020 18A
19F8: 2020202020202020 100
1A00: 0F102020EC20207B 206
1A08: FA76FB7C20202020 367
1A10: 20E1462E472E542E 26C
1A18: 482EE32020292B20 20D
1A20: 2035373937352020 171
1A28: 292B202020202020 114
1A30: 00020320202020EA 16F
1A38: EBD824271D212427 297
1A40: 1D2124271D212427 112
1A48: 1D2124271D212427 112
1A50: 1D2124D5ED777878 38B
1A58: 77EDD8C9CCCFD2C6 638
1A60: C9CCCFD2C6C9CCCF 660
1A68: D2C6C9CCCFD2C6C9 65D
1A70: CCCFD2C6C9D52020 511
1A78: 40E2E2E2E2E2E2E2 66E
1A80: E2E2E2E2E2E2E240 66E
1A88: 20EAEB20F0F1F0F1 5D7
1A90: 208C9F9220202020 24D
1A98: 20B4B7B9BC202020 360
1AA0: 7DED7AED7E202020 3AF
1AA8: 9BF6F6F6F6F6F6F6 755
1AB0: F6F6F6F6F69D2020 5AB
1AB8: 2020202020952020 175
1AC0: 2020202020202020 100
1AC8: 9BEEEEEEEE9D2020 530
1AD0: 2020202020202020 100
1AD8: 952020202020BEBF 2B2
1AE0: C1C3202020207DFA 37B
1AE8: ECFB7E2020F0F106 48C
1AF0: 20F0F12020202000 281
1AF8: 020320207F20200C 110
1B00: 2020202020202020 100
1B08: 2020202080848488 290
1B10: 2020202020202020 100
1B18: 2020202020202020 100
1B20: 2020722020202020 152
1B28: 2020202020202000 0E0
1B30: 0101040520202020 08B
1B38: 2020202020202020 100
1B40: 2020202020202020 100
1B48: 8084848488202020 2F4
1B50: 2020702020202020 150
1B58: 20DAF2F2F2F2F2F2 6A6
1B60: F23A3B3CF2F2F2F2 56B
1B68: F2F2F2F2F2F2F2F2 790
1B70: F2F2F2F2F2F2DD20 6A9
1B78: 2020F7F7F7F7F7F7 60A
1B80: F7F7F7F7F7F72020 60A
1B88: 2020202020202020 100
1B90: 2020202020202020 100
1B98: 2020202020202020 100
1BA0: 2020202020202020 100
1BA8: 202020A4A7202020 20B
1BB0: AFB1202020202020 220
1BB8: 2020202020202020 100
1BC0: 2020202020206B20 14B
1BC8: 2020202020202020 100
1BD0: 8084848820202020 290
1BD8: 2020202020202020 100
1BE0: 2020202020202020 100
1BE8: 207220205C3E6007 1F3
1BF0: 090B692020200001 0DE
1BF8: 0104052020202020 0AA
1C00: 2020208084848488 2F4
1C08: 20202020882078289 294
1C10: 2020201113151719 0C9
1C18: 20206A2020202020 14A
1C20: 6F20202020202020 14F
1C28: 202020206D202020 14D
1C30: 2020201113151719 0C9
1C38: 2020202020808484 228
1C40: 8484882020202020 230
1C48: 83FAFB8389202020 3E4
1C50: 2020202020206B20 14B
1C58: DBF3F3F3F3F3F3F3 780
1C60: F3F3F3F3F3F3F3F3 798
1C68: F3F3F3F3F3737475 61B
1C70: F3F3F3F3F3F3F3DE 783
1C78: 20F8F8F8F8F8F8F8 6E8
1C80: F8F8F8F8F8F8F820 6E8
1C88: 20202020206F2020 14F
1C90: 202020206F202020 14F
1C98: 2020209E9F9F9FA0 37B
1CA0: A19F9F9F9FA22020 3FF
1CA8: 2020A3A5A8AAACAE 434
1CB0: B020202020202080 1F0
1CB8: 8484882020202020 230
1CC0: 2020202020202020 100
1CC8: 2020202020202020 100
1CD0: EC86EC8920202020 367
1CD8: 20202020209E9F9F 27C
1CE0: 9F9F9FA220202020 2FF
1CE8: 2020205B5D5F6108 1E0
1CF0: 0A68202020202020 132
1CF8: 2020202020206F20 14F
1D00: 2020208185858189 2F5
1D08: 20206C2020202020 14C
1D10: 202020121416181A 0CE
1D18: 2020202020712020 151
1D20: 20202020206E2020 14E
1D28: 206C202020202020 14C
1D30: 202020121416181A 0CE
1D38: 6A2020202052454C 1CD
1D40: 4158892020202020 1C2
1D48: 20202020206E2020 14E
1D50: 6C20202020202020 14C
1D58: 20DCF4F4737475F4 534
1D60: F4F4F4F4F4F4F4F4 7A0
1D68: F4F4F4F4F4F4F4F4 7A0
1D70: F4F4F4F4F4F4DF20 6B7
1D78: F9F9F9F9F9F9F9F9 7C8
1D80: F9F9F9F9F9F9F9F9 7C8
```

(continued on page 87)

(continued from page 85)

```
1D88: 202071202020206D 19E
1D90: 2020202020207020 150
1D98: 2020202020202020 100
1DA0: 2020202020202020 100
1DA8: 202020A6A9ABAD20 327
1DB0: 202020206A202082 1AC
1DB8: 8782892020209E9F 32F
1DC0: 9FA0A19F9FA22020 400
1DC8: 2020202020202020 100
1DD0: 20202020206A2020 14A
1DD8: 2020202020202020 100
1DE0: 202020202020206E 14E
1DE8: 2020202020202020 100
1DF0: 202020206C202020 14C
1DF8: 2071202020202020 151
1E00: 0707070202020202 01F
1E08: 0404040202020202 016
1E10: 0202020201020202 00F
1E18: 0202020202020201 00F
1E20: 0203020202030202 012
1E28: 0202020202020202 010
1E30: 0202020202010202 00F
1E38: 0202010707070707 028
1E40: 0707070707070707 038
1E48: 0707070707070707 038
1E50: 0707070702010909 031
1E58: 0102010101010101 009
1E60: 0101010101010101 008
1E68: 0101010101010101 008
1E70: 0101010101010101 008
1E78: 0606060606060606 030
1E80: 0606060606060606 030
1E88: 0202020102020202 00F
1E90: 0203030302020201 012
1E98: 0207070707020202 024
1EA0: 0202020202020202 010
1EA8: 0401010101010101 00B
1EB0: 0101010101040202 00D
1EB8: 0201020202070702 019
1EC0: 0202010101010202 00C
1EC8: 0903030303090202 022
1ED0: 0202020302020207 016
1ED8: 0702020202020202 015
1EE0: 0202020202020202 010
1EE8: 0402020202020202 012
1EF0: 0202020202020202 010
1EF8: 0202020202020202 010
1F00: 0707070209020202 026
1F08: 0404040202020202 016
1F10: 0207070707070707 033
1F18: 0707070202050502 025
1F20: 0203030303030202 015
1F28: 0505020202020202 016
1F30: 0202020202010105 011
1F38: 0502070707070707 031
1F40: 0707070707070707 038
1F48: 0707070707070707 038
1F50: 0707070704040404 02C
1F58: 0404010101010101 00E
1F60: 0101010101010101 008
1F68: 0101010101010101 008
1F70: 0101010101010101 008
1F78: 0607070707070707 037
1F80: 0707070707070706 037
1F88: 0204040202020202 014
1F90: 0203030302020202 013
1F98: 0207070707020202 024
1FA0: 0209030902020202 01F
1FA8: 0401010101010101 00B
1FB0: 0101010101040202 00D
1FB8: 0202020202070402 017
1FC0: 0202020102020202 00F
1FC8: 0903030303090202 022
1FD0: 0202020202020204 012
1FD8: 0702020202020303 017
1FE0: 0303020201020204 013
1FE8: 0F04020202020202 01F
1FF0: 0202020202020202 010
1FF8: 0202020202020202 010
2000: 0707070209020204 028
2008: 0304030402020202 016
2010: 0207050505050505 027
2018: 0505070202020202 01B
2020: 0203030303030202 015
2028: 0202020202020202 010
2030: 0E0E0E0202020205 037
2038: 0507070707070707 036
2040: 0707070707070707 038
2048: 0707070707070707 038
2050: 070707070C0C0C0C 04C
2058: 0C0C010101010101 01E
2060: 0101010101010101 008
2068: 0101010101010101 008
2070: 0101010101010101 008
2078: 0606060606060606 030
2080: 0606060606060606 030
2088: 0204040209090909 030
2090: 0203030302020202 013
2098: 0207070707020202 024
20A0: 0109010901020202 01B
20A8: 0401010101010101 00B
20B0: 0101010101040202 00D
20B8: 0202020202070202 015
20C0: 0202020202020202 010
20C8: 0903030303090202 022
20D0: 0202020202020202 010
20D8: 0702020202020707 01F
20E0: 0707020202020407 021
20E8: 0F07040202090900 030
20F0: 0209090202020208 024
20F8: 080802020F020207 02E
2100: 0707070202020202 01F
2108: 0404040203030303 01A
2110: 0202020202020202 010
2118: 0202020202020202 010
2120: 0202050202020202 013
2128: 020202020202020E 01C
2130: 0E0E0E0E02020202 040
2138: 0202020202020202 010
2140: 0202020202020202 010
2148: 0606060606020202 024
2150: 0202000202020202 00E
2158: 0206060606060606 02C
2160: 0606060606060606 030
2168: 0606060606060606 030
2170: 0606060606060606 030
2178: 0000000000000000 000
2180: 0000000000000000 000
2188: 0202020202020202 010
2190: 0202020202020202 010
2198: 0202020202020202 010
21A0: 0202020202020202 010
21A8: 0202020202020202 010
21B0: 0202050505050202 01C
21B8: 0202020202020202 010
21C0: 0202020202020602 014
21C8: 0203030303020202 014
21D0: 0707070702020202 024
21D8: 0202020202020202 010
21E0: 0202020202020202 010
21E8: 0200020000000000 004
21F0: 0000000202020808 016
21F8: 0808080202020202 022
2200: 0707070202020202 01F
2208: 0404040209090903 02C
2210: 0202020606060606 024
2218: 0202000202020202 00E
2220: 0002020202020202 00E
2228: 0202020207020202 015
2230: 0202020000000000 006
2238: 0202020202020202 010
2240: 0202020202020202 010
2248: 0000080806020202 01C
2250: 0202020202020202 010
2258: 0606060606060606 030
2260: 0606060606060606 030
2268: 0606060606060606 030
2270: 0606060606060606 030
2278: 0000000000000000 000
2280: 0000000000000000 000
2288: 0202020202000202 00E
2290: 0202020202020202 010
2298: 0202020000000002 008
22A0: 0200000000000202 006
22A8: 0202020202020202 010
22B0: 0202050505050203 01D
22B8: 0303030202020202 013
22C0: 0202020202020202 010
22C8: 0203030303020202 014
22D0: 0F0F0F0702020202 03C
22D8: 0202020202010101 00D
22E0: 0101010102020202 00C
22E8: 0202020000000000 006
22F0: 0000000202020202 00A
22F8: 0202020202020002 00E
2300: 0707070202020202 01F
2308: 0404060202020202 018
2310: 0202020606060606 024
2318: 0202020202040202 012
2320: 0202020202060202 014
2328: 0207020202020202 015
2330: 0202020000000000 006
2338: 0202020202060606 01C
2340: 0606020202020202 018
2348: 0202020202020202 010
2350: 0302020202020202 011
2358: 0206060606060606 02C
2360: 0606060606060606 030
2368: 0606060606060606 030
2370: 0606060606060606 030
2378: 0000000000000000 000
2380: 0000000000000000 000
2388: 0202020202020202 010
2390: 0202020202020402 012
2398: 0202020202020202 010
23A0: 0202020202020202 010
23A8: 0202020202020202 010
23B0: 0202050500050208 01D
23B8: 0808030202020202 01D
23C0: 0206060202020202 018
23C8: 0203030303020202 014
23D0: 0202020202010202 00F
23D8: 0202020202020202 010
23E0: 0202020202020201 00F
23E8: 0202020000000000 006
23F0: 0000000202020202 00A
23F8: 0206020202020202 014
2400: 03C0000F600007B0 1E9
2408: 0007D80003EC0041 20F
2410: 0FE017FFB0470F9E 3A9
2418: 16F7F841EFE003C0 4D8
2420: 000780000F000000 096
2428: 0000000000000000 000
2430: 0000000000000000 000
2438: 0000000000000000 000
2440: 03C0000F600007B0 1E9
2448: 0007D80003EC0021 1EF
2450: 0FE08FFFB0270F9E 401
2458: 8EF7F821EFE003C0 530
2460: 000780000F000000 096
2468: 0000000000000000 000
2470: 0000000000000000 000
2478: 0000000000000000 000
2480: 0003C00006F0000D 1C6
2488: E0001BE00037C007 2D9
2490: E0820DFFE879F0E2 5A1
2498: 1FEF6807F7820003 2F9
24A0: C00001E00000F000 291
24A8: 0000000000000000 000
24B0: 0000000000000000 000
24B8: 0000000000000000 000
24C0: 0003C00006F0000D 1C6
24C8: E0001BE00037C007 2D9
24D0: E0840DFFF179F0E4 5AE
24D8: 1FEF7107F7840003 304
24E0: C00001E00000F000 291
24E8: 0000000000000000 000
24F0: 0000000000000000 000
24F8: 0000000000000000 000
2500: F0000FF0000FF000 2EE
2508: 0FF8001F1C7E380E 206
2510: FF7007FFE00318C0 430
2518: 0718E00718E007FF 304
2520: E007E7E006FF6003 416
2528: 00C007FFE00EFF70 423
2530: 1C7E38F8001FF000 2D9
2538: 0FF0000FF0000F00 20D
2540: 00000078001E7800 10E
2548: 1E78001E7C7E3E0E 1FA
2550: FF7007FFE00318C0 430
2558: 0718E00718E007FF 304
2560: E007E7E006FF6003 416
2568: 00C007FFE00EFF70 423
2570: 7C7E3E78001E7800 246
2578: 1E78001E00000000 0B4
2580: 0000000000003C00 03C
2588: 3C3C003C3C7E3C3E 1E8
2590: FF7C07FFE00318C0 43C
2598: 075AE00718E007FF 346
25A0: E007E7E0067E6003 395
25A8: 00C007FFE03EFF7C 45F
25B0: 3C7E3C3C003C3C00 1AA
25B8: 3C00000000000000 03C
25C0: 0000000007C0000F 0D6
25C8: C0000F00000F0000 0DE
25D0: 1E00003C00003C00 096
25D8: 00780000F00000F0 258
25E0: 0003F00003E00000 1D6
25E8: 0000000000000000 000
25F0: 0000000000000000 000
25F8: 0000000000000000 000
2600: 0000000000C00000 0C0
2608: C00003C00003C000 246
2610: 1FC0003F8001FC00 29B
2618: 03F80003C00003C0 281
2620: 0003000003000000 006
2628: 0000000000000000 000
2630: 0000000000000000 000
2638: 0000000000000000 000
2640: 0000000000000000 000
2648: 000000000000000F 00F
2650: FFF00FFFF00FFFF0 5EB
2658: 0FFFF00000000000 1FE
2660: 0000000000000000 000
2668: 0000000000000000 000
2670: 0000000000000000 000
2678: 0000000000000000 000
2680: 0000000300000300 006
2688: 0003C00003C00003 189
2690: F80001FC00003F80 2B4
2698: 001FC00003C00003 1A5
26A0: C00000C00000C000 240
26A8: 0000000000000000 000
26B0: 0000000000000000 000
26B8: 0000000000000000 000
26C0: 00000003E00003F0 1D6
26C8: 0000F00000F00000 1E0
26D0: 7800003C00003C00 0F0
26D8: 001E00000F00000F 03C
26E0: 00000FC00007C000 196
26E8: 0000000000000000 000
26F0: 0000000000000000 000
26F8: 0000000000000000 000
2700: 003C00003C00003C 0B4
2708: 00003C00003C0000 078
2710: 3C00003C00003C00 0B4
2718: 003C00003C00003C 0B4
2720: 00003C00003C0000 078
2728: 3C00000000000000 03C
2730: 0000000000000000 000
2738: 0000000000000000 000
2740: 0018000199800181 1B4
2748: 800C00300C003000 0F8
2750: 000030000C30000C 078
2758: 0000006000066000 0C6
2760: 0600000030000C30 072
2768: 000C0000000C0030 048
2770: 0C00300181800199 1D8
2778: 8000180000000000 098
2780: 0000000018000199 0B2
2788: 8001818006006006 1EE
2790: 0060180018180018 0C0
2798: 00000030000C3000 06C
27A0: 0C00000018001818 054
27A8: 0018060060060060 0E4
27B0: 0181800199800018 234
27B8: 0000000000000000 000
27C0: 0000000000000018 018
27C8: 0001998001818000 21C
27D0: 00000C00300C0030 078
27D8: 0000001800181800 048
27E0: 180000000C00300C 060
27E8: 0030000000018180 132
27F0: 0199800018000000 132
27F8: 0000000000000000 000
2800: 0000000000000000 000
2808: 0000180000DB0000 0F3
2810: C3000300C00300C0 249
2818: 0000000600600600 06C
2820: 600000000300C003 126
2828: 00C000C30000DB00 25E
2830: 0018000000000000 018
2838: 0000000000000000 000
2840: 0000000000000000 000
2848: 0000000000000000 000
2850: 180000DB0000C300 1B6
2858: 0000000181800181 184
2860: 8000000000C30000 143
2868: DB00001800000000 0F3
2870: 0000000000000000 000
2878: 0000000000000000 000
2880: 0000000000000000 000
2888: 0000000000000000 000
2890: 0000000000006600 066
2898: 0066000000000000 066
28A0: 0000660000660000 0CC
28A8: 0000000000000000 000
28B0: 0000000000000000 000
28B8: 0000000000000000 000
28C0: 0000000000000000 000
28C8: 0000000000000000 000
28D0: 0000000000000000 000
28D8: 001800003C00003C 090
28E0: 0000180000000000 018
28E8: 0000000000000000 000
28F0: 0000000000000000 000
28F8: 0000000000000000 000
2900: 0000000000000000 000
2908: 0000000000000000 000
2910: 0000000000000000 000
2918: 0000000018000018 030
2920: 0000000000000000 000
2928: 0000000000000000 000
2930: 0000000000000000 000
```

(continued on next page)

(continued from previous page)

```
2938: 0000000000000000 000
2940: 700000F8C00098E0 3A0
2948: 005BF0003FF8000D 28F
2950: BC000EFE005F6E00 295
2958: 7B780007FC0036FF 32B
2960: 001C77C00039F000 27C
2968: 3C30001C30000E00 0C6
2970: 0007800003E00000 16A
2978: 6000006000000000 0C0
2980: 700000F8000098E0 2E0
2988: 001BF0003F78000D 1CF
2990: FC000EDE00DF7E00 345
2998: 7B780007FC002EFF 323
29A0: 003C77800039E000 24C
29A8: 3C70001C60000E00 136
29B0: 0007000003C00000 0CA
29B8: E00000C000000000 1A0
29C0: F000007800007C00 1E4
29C8: 003E00001F80001F 0FC
29D0: FFF87F77CC1F77C6 515
29D8: 1FFFFF7BF7FF07EF 584
29E0: FE1F800000000000 19D
29E8: 0000000000000000 000
29F0: 0000000000000000 000
29F8: 0000000000000000 000
2A00: F000007800006C00 1D4
2A08: 003600001B80001F 0F0
2A10: FFF87F77CC1F77C6 515
2A18: 1FFFFF7BF7FF07EF 584
2A20: FE1F800000000000 19D
2A28: 0000000000000000 000
2A30: 0000000000000000 000
2A38: 0000000000000000 000
2A40: 07FFE00FFFF00E3C 42E
2A48: 700E3C700E3C700F 1F3
2A50: FFF00F7EF006F81F6 552
2A58: E7FFE7F8001F9E00 482
2A60: 790781E001E78000 349
2A68: 3E00007C0001E780 222
2A70: 0781E01E00787C00 27A
2A78: 3E70000E30000C00 0F8
2A80: 07FFE00FFFF00E3F 431
2A88: F00E3C700E3C700F 273
2A90: FFF00F7EF00F81F0 4EC
2A98: 37FFEC70001E7E00 336
2AA0: 7E0781E001E78000 34E
2AA8: 3E00007C0001E780 222
2AB0: 0781E03E007C3C00 25E
2AB8: 3C38001C0000000 090
2AC0: 07FFE00FFFF00E3F 431
2AC8: F00E3FF00E3C700F 2F6
2AD0: FFF00F7EF00F81F0 4EC
2AD8: 07FFE038001C3E00 278
2AE0: 7C3781EC01E78000 388
2AE8: 3E00007C0001E780 222
2AF0: 1F81F81E00781C00 24A
2AF8: 3800000000000000 038
2B00: 07FFE00FFFF00E3F 431
2B08: F00E3FF00E3FF00F 379
2B10: FFF00F7EF00F81F0 4EC
2B18: 07FFE01C00381E00 258
2B20: 781F81F801E78000 378
2B28: 3E00007C000FE7F0 2A0
2B30: 0F81F00E00700000 1FE
2B38: 0000000000000000 000
2B40: 001800003C0000E6 13A
2B48: 0007C600007E0000 14B
2B50: 6600007E00066600 150
2B58: 07FE0006E60000DB 2CC
2B60: 0001E700019B8003 207
2B68: 67800399C00666C0 36F
2B70: 0799E00E66600FFF 362
2B78: E000000000000000 0E0
2B80: 001800003C0000F6 14A
2B88: 0003EE00007E0000 16F
2B90: 7E00006600027E00 164
2B98: 03FE0002E60000DB 2C4
2BA0: 0001E700019B8003 207
2BA8: 67800399C00666C0 36F
2BB0: 0799E00E66600FFF 362
2BB8: E000000000000000 0E0
2BC0: 003C0000FF000381 1BF
2BC8: C00700E00E00700C 231
2BD0: 00300C00300DE7B0 210
2BD8: 0D24B00D65B00DE7 2F7
2BE0: B006186006186006 1B2
2BE8: 18600300C003FF80 2BD
2BF0: 01C38000FF00007E 2C1
2BF8: 0000180000000000 018
2C00: 003C0000FF000381 1BF
2C08: C00700E00E00700C 231
2C10: 00300C00300DE7B0 210
2C18: 0D65B00D24B00DE7 2F7
2C20: B006186006186006 1B2
2C28: 18600300C003FF80 2BD
2C30: 01C38000E700007E 2A9
2C38: 0000180000000000 018
2C40: 003C0000FF000381 1BF
2C48: C00700E00E00700C 231
2C50: 00300C00300DE7B0 210
2C58: 0DA6B00D24B00DE7 338
2C60: B006186006186006 1B2
2C68: 18600300C003FF80 2BD
2C70: 01C38000FF00007E 2C1
2C78: 0000180000000000 018
2C80: 003C0000FF000381 1BF
2C88: C00700E00E00700C 231
2C90: 00300C00300DE7B0 210
2C98: 0D24B00DA6B00DE7 338
2CA0: B006186006186006 1B2
2CA8: 18600300C003FF80 2BD
2CB0: 01C38000E700007E 2A9
2CB8: 0000180000000000 018
2CC0: 003C0003FFC03FFF 33C
2CC8: FCFE667FFFFFFFF9 6D5
2CD0: 999FFFFFFF3FFFFC 66F
2CD8: 0000000000000000 000
2CE0: 0000000000000000 000
2CE8: 0000000000000000 000
2CF0: 0000000000000000 000
2CF8: 0000000000000000 000
2D00: 003C0003FFC03FFF 33C
2D08: FCF9999FFFFFFFFE 728
2D10: 667FFFFFFF3FFFFC 61C
2D18: 0000000000000000 000
2D20: 0000000000000000 000
2D28: 0000000000000000 000
2D30: 0000000000000000 000
2D38: 0000000000000000 000
2D40: 003C0003FFC03FFF 33C
2D48: FCFF3CFFF9FF9FFF 6CC
2D50: 3CFFFFFFFF3FFFFC 672
2D58: 0000000000000000 000
2D60: 0000000000000000 000
2D68: 0000000000000000 000
2D70: 0000000000000000 000
2D78: 0000000000000000 000
2D80: 000CFC00C3FF0C0C 2E2
2D88: FF00C3FFC00CFF0C 498
2D90: C3FFC00CFF00C3FF 54F
2D98: 0C0CFF00C3FF000C 2E5
2DA0: FC00000000000000 0FC
2DA8: 0000000000000000 000
2DB0: 0000000000000000 000
2DB8: 0000000000000000 000
2DC0: 0003FC000CFF00C3 2CD
2DC8: FFC0CFF00C3FF30 4C8
2DD0: 0CFF00C3FFC0C0CFF 4A4
2DD8: 00C3FF000CFF0003 2D0
2DE0: FC00000000000000 0FC
2DE8: 0000000000000000 000
2DF0: 0000000000000000 000
2DF8: 0000000000000000 000
2E00: 00C300000000003C 0FF
2E08: 00000000000C3000 0C3
2E10: 0000003C000300C0 0FF
2E18: 00000000CCC00000 18C
2E20: 0003330000CCC003 1C5
2E28: FFC003FFC003FFC0 543
2E30: 03FFC003FFC000FF 483
2E38: 0000000000000000 000
2E40: 00660000000000C3 129
2E48: 00000000003C0003 03F
2E50: 00C000C300000000 183
2E58: 03330000000000CC 102
2E60: C000000003330003 0F9
2E68: FFC003FFC003FFC0 543
2E70: 03FFC003FFC000FF 483
2E78: 0000000000000000 000
2E80: 3F3000FFC300FF30 360
2E88: 30FFC300FF3003FF 423
2E90: C330FF3003FFC300 3E7
2E98: FF3030FFC3003F30 390
2EA0: 0000000000000000 000
2EA8: 0000000000000000 000
2EB0: 0000000000000000 000
2EB8: 0000000000000000 000
2EC0: 3FC000FF3000FFC3 3F0
2EC8: 00FF3033FFC300FF 423
2ED0: 300CFFC300FF3033 360
2ED8: FFC300FF30003FC0 3F0
2EE0: 0000000000000000 000
2EE8: 0000000000000000 000
2EF0: 0000000000000000 000
2EF8: 0000000000000000 000
2F00: 7FFFFEFFDE7FFF9D 674
2F08: BFFF5D7FFEDE3FFD 5B2
2F10: DDDFFBDBEFF7D603 651
2F18: EFCDEFDFDE1F7FFF 605
2F20: FE00000000000000 0FE
2F28: 0000000000000000 000
2F30: 0000000000000000 000
2F38: 00000000000000FF 0FF
2F40: 0000C80003900006 161
2F48: 24000C590038AA00 16B
2F50: 611401CA89031143 220
2F58: 0622A60CD14C1128 230
2F60: 98251530CAAA6019 2EF
2F68: 44C0A4A180525300 36E
2F70: 292600140C000A18 091
2F78: 0005300002600000 097
2F80: 0000000000000000 000
2F88: 0000000000000000 000
2F90: 0000000000000000 000
2F98: 0000000000000000 000
2FA0: 0000000000000000 000
2FA8: 0000000000000000 000
2FB0: 0000000000000000 000
2FB8: 0000000000000000 000
2FC0: 7FFFFEFFFFFFC10D 647
2FC8: 07DD7577DD757FDD 47E
2FD0: 0D7FDD6D7FDD7577 41E
2FD8: C17507FFFFFF7FFF 5B8
2FE0: FE00000000000000 0FE
2FE8: 0000000000000000 000
2FF0: 0000000000000000 000
2FF8: 00000000000000FF 0FF
3000: 7FFFFEFDFE0FF27D 3F5
3008: EFE7BA1FF9B43FFD 598
3010: 6BFFFB100FFBFFF7 575
3018: FC0037FEBF6FFD7E 4DA
3020: DFFAFDBF7FFFFE00 611
3028: 0000000000000000 000
3030: 0000000000000000 000
3038: 00000000000000FF 0FF
3040: F9FFFF8989990FEF 620
3048: F9898989F9F98F00 515
3050: 00007FFFFEFFFFFF 579
3058: E6F041DEF75DDE97 5BE
3060: 5D869041DE9F7DDE 48C
3068: FF7D86F041FFFFFF 630
3070: 7FFFFE0000000000 27C
3078: 00000000000000FF 0FF
3080: FFFFFEFFFFC3DAAA 741
3088: FEFFFFC3DAAAFEFF 740
3090: FFC3DAAAFEFFFFC3 705
3098: DAAAFEFFFFC3FFFF 741
30A0: FE0000007DE7DF45 386
30A8: 145145E7DF4D1450 321
30B0: 7D14500600000000 0E7
30B8: 00000000000000FF 0FF
30C0: 7FFFFEC18303DEBD 55E
30C8: 7BDEBD7FDEBD7FC1 570
30D0: 837FDEBD7FDEBD7F 536
30D8: DEBD7BC183037FFF 4DB
30E0: FE00000000000000 0FE
30E8: 0000000000000000 000
30F0: 0000000000000000 000
30F8: 00000000000000FF 0FF
3100: 001800003C00003C 090
3108: 0000180001E78003 183
3110: E7C006FF6006DB60 44D
3118: 06DB6006C360035A 2C7
3120: C0015A8000FF0000 29A
3128: E70000E70000E700 2B5
3130: 00E70000E70000E7 2B5
3138: 0000C30001818000 1C5
3140: 7FFFE0FFFFF8C3E6 6FD
3148: 6CDFE666C2FFFEFA 650
3150: FFFFC2F01FFEF6C7 68A
3158: FE101FFFFFFE7FFF 5A7
3160: FC00000000000000 0FC
3168: 0000000000000000 000
3170: 0000000000000000 000
3178: 00000000000000FF 0FF
3180: FFFFFFFFFFFFD555 724
3188: 1FFFFFFFD5555FFF 5A4
3190: FFDFD5551FFFFFFF 624
3198: CD551FFFFFFFD505 518
31A0: 5FFFFFFF00000020 37C
31A8: 001F2003D1F9F011 30D
31B0: 2003D120001F0000 133
31B8: 00000000000000FF 0FF
31C0: 7FFFFEFFFFFFFFFF 777
31C8: FFFFFFFFFFFFFFFF 7F8
31D0: FFFFFFFFFFFFFFFF 7F8
31D8: FFFFFFFFFFFF7FFF 778
31E0: FE00000000000000 0FE
31E8: 0000000000000000 000
31F0: 0000000000000000 000
31F8: 00000000000000FF 0FF
3200: 2046495245001532 18D
3208: C28846535449434B 30E
3210: 20525453002A321C 191
3218: 8953455452454720 273
3220: 4C44582023464952 20C
3228: 4500353226894C44 1EB
3230: 412023300045322B 156
3238: 8953544120494E45 26D
3240: 5254494100553230 1E7
3248: 8956484B20535441 27A
3250: 20302C58005D323A 19D
3258: 89494E58006F3244 25D
3260: 8943505820234D55 259
3268: 4C544950AA39007B 297
3270: 324E89424E452056 254
3278: 484B000632588940 278
3280: 4441202334009332 1C1
3288: 62895354412056AA 2F3
3290: 343000A0326C8953 27E
3298: 54412056AA343100 21A
32A0: AB3276894C444120 2CD
32A8: 233100B832808953 29A
32B0: 54412056AA323100 218
32B8: C532858953544120 30D
32C0: 56AA323900D3328A 2FA
32C8: 8953544120444952 270
32D0: 454300E432C68953 340
32D8: 544120434F554E54 23E
32E0: 31AA3300F532D089 30E
32E8: 53544120434F554E 23D
32F0: 5431AA34000633DA 276
32F8: 8953544120434F55 278
3300: 4E5431AA35001733 1FC
3308: E48953544120434F 307
3310: 554E5431AA36002A 232
3318: 33EE894C44582050 302
3320: 524553504C415945 265
3328: 52003D33F8894C44 2D3
3330: 41204C4556454C31 20A
3338: 55502C5800453302 1A3
3340: 8A53454300503307 1EF
3348: 8A53424320233100 1D6
3350: 5A330C8A4C535220 234
3358: 41006233118A5441 206
3360: 59007533168A4C44 231
3368: 41204C4556454C43 21C
3370: 4F4C2C590085331B 1F3
3378: 8A53544120455854 283
3380: 5241AA340090331E 252
3388: 8A4C445820233100 1E6
3390: A333208A4C444120 271
3398: 50524553504C4159 270
33A0: 455200AE332A8A43 26F
33A8: 4D5020233000BA33 1FD
33B0: 348A42455120ACAA 30C
33B8: 3400C6333E8A4C44 285
33C0: 5820233239000D333 20C
33C8: 488A4C4441202332 218
33D0: 323100E233528A53 2A7
33D8: 544120313036342C 1AC
33E0: 5800F133578A5354 304
33E8: 4120313037332C58 1B0
33F0: 00FC335C8A4C4441 2E6
33F8: 202334000C34668A 1A7
3400: 5354412035353333 1D8
3408: 362C58001C346B8A 1FF
3410: 5354412035353334 1D9
3418: 352C580029347 08A 210
3420: 4C44412023313438 1B1
3428: 003634718A535441 24D
3430: 2032303433004334 160
3438: 728A535441203230 266
3440: 34340050347308A53 23C
3448: 5441203230343500 180
3450: 5D34748A53544120 297
3458: 32303436006A347A 1E4
3460: 8A4C444120233130 1FF
3468: 300077347F8A5354 28B
3470: 412059B9AA330084 2D4
3478: 34848A5354412059 2A3
3480: B9AA340091348 98A 36F
3488: 4C44412023313530 1AA
3490: 009E348E8A535441 2D2
3498: 2059B9AA3500AB34 2F0
34A0: 938A5354412059B9 337
34A8: AA3600B634988A4C 338
34B0: 444120233200C734 1F5
34B8: 9D8A535441204849 2C0
34C0: 474848B9AA3300D2 33F
34C8: 34A28A4C44412023 274
34D0: 3300E334A78A5354 322
34D8: 4120484947484889 282
34E0: AA3400EE34AC8A4C 382
34E8: 444120233500FF34 230
```

Table 1. Examples of use of VDU 23, 2,

	Text parameters				Graphics parameters				
	Text area		Text colours		Graphics area		Graphics colours		
	Start	End	Fore-ground	Back-ground	Start	End	Fore-ground	Back-ground	
VDU 23, 2,	TXS,	TXF,	TFC,	TBC,	GCS,	GCF,	GFC,	GBC	command
VDU 23, 2,	0,	24,	129,	130,	255,	0,	0,	0	
	Start at line 0 end at line 24 i.e. cover all screen with text colour		Red foreground	Green background	Ignore graphics parameters — no part of screen is given graphics colouring				
VDU 23, 2,	255,	0,	0,	0,	0,	24,	145,	151	
	Ignore text parameters no part of screen will have text colours — all screen can be used for graphics				Start at line 0 end at line 24 i.e. cover all screen in graphics colour		Red fore ground	White background	
VDU 23, 2,	0,	4,	131,	132,	5,	24,	146,	147	
	Start text at line 0 and end at line 4 —small text area for messages		Yellow fore-ground	Blue back-ground	Start graphics at 5 end at 24 — most of screen is for graphics		Green fore-ground	Yellow back-ground	
VDU 23, 2,	0,	24,	131,	0,	255,	0,	0,	0	
	Start at line 0 end at line 24 — text covers all screen		Yellow fore-ground	Default back-ground colour i.e. black	Ignore graphics parameters				

Teletext colour control codes.

Text colours		Graphics colours
129	Red	145
130	Green	146
131	Yellow	147
132	Blue	148
133	Magenta	149
134	Cyan	150
135	White	151

BBC

Chris Bowerman's routine allows graphics in Modes 3, 6 and 7.



Table 2.

Machine code routine to read status of pixel stored at PX and PX + 1 and PY.

```
PX = &8D:PY = &8F
[...
.readpoint
LDA # &60:STAparameter:LDA # 4:
STAparameter + 1:LDA PX:JSRpoke
INCparameter:LDA PX + 1:JSRpoke
LDA # &64:
STAparameter:LDAPY:JSRpoke
JSRpoint:RTS
  A contains state of pixel
...]
```

1. Store the low byte of the X co-ordinate at PX.
2. Store the high byte of the X co-ordinate at PX + 1.
3. Store the Y co-ordinate at PY.
4. A jump to subroutine 'readpoint' — i.e. JSR readpoint — will return the status of the pixel — 0,1, or − 1 — in the accumulator.

In Modes 3, 6 and 7 on the BBC Micro and Electron you can't use graphics commands. Modes 3 and 6 would be much more attractive if graphics were available, especially if there were no ugly stripes between lines. The use of graphics in Mode 7 requires tedious calculations and the awkward manipulation of teletext codes — a plot command would make things much easier.

With this in mind I decided to provide some graphics commands for these modes. The commands automatically adjust for the different modes and are Tube-compatible.

The screen resolutions are as follows:

Mode	X range	Y range
3	0 to 639	0 to 199
6	0 to 319	0 to 199
7	0 to 73	0 to 74

Mode 3 thus offers a similar resolution to Mode 0, and Mode 6 offers a similar resolution to Mode 4 but both at a saving of 4K. The new commands will accept any of the usually allowed variables as parameters.

PLOT 69,X,Y

sets pixels at given co-ordinates to current foreground colour. Unplotting is achieved by plotting in the background graphics colour which should always be

GCOL,0

The foreground graphics colour may be changed by means of the usual GCol command — see page 262 of the user guide.

GCOLx,1

sets foreground graphics colour and

GCOLx,0

sets background graphics colour.

GCOL0,x

sets ordinary plot mode,

GCOL1,x

sets Or plot mode,

GCOL2,x

sets And plot mode, and

GCOL3,x

sets EOr plot mode, and

GCOL4,x

sets inverse mode. The Colour command must not be used to change actual screen colours but the VDU19 command should be used if the new Plot commands are to work properly.

In Mode 7 *FX1,x determines whether Plotting will set or clear pixels. Plotting after *FX1,1 will set pixels and Plotting after *FX1,0 will clear them.

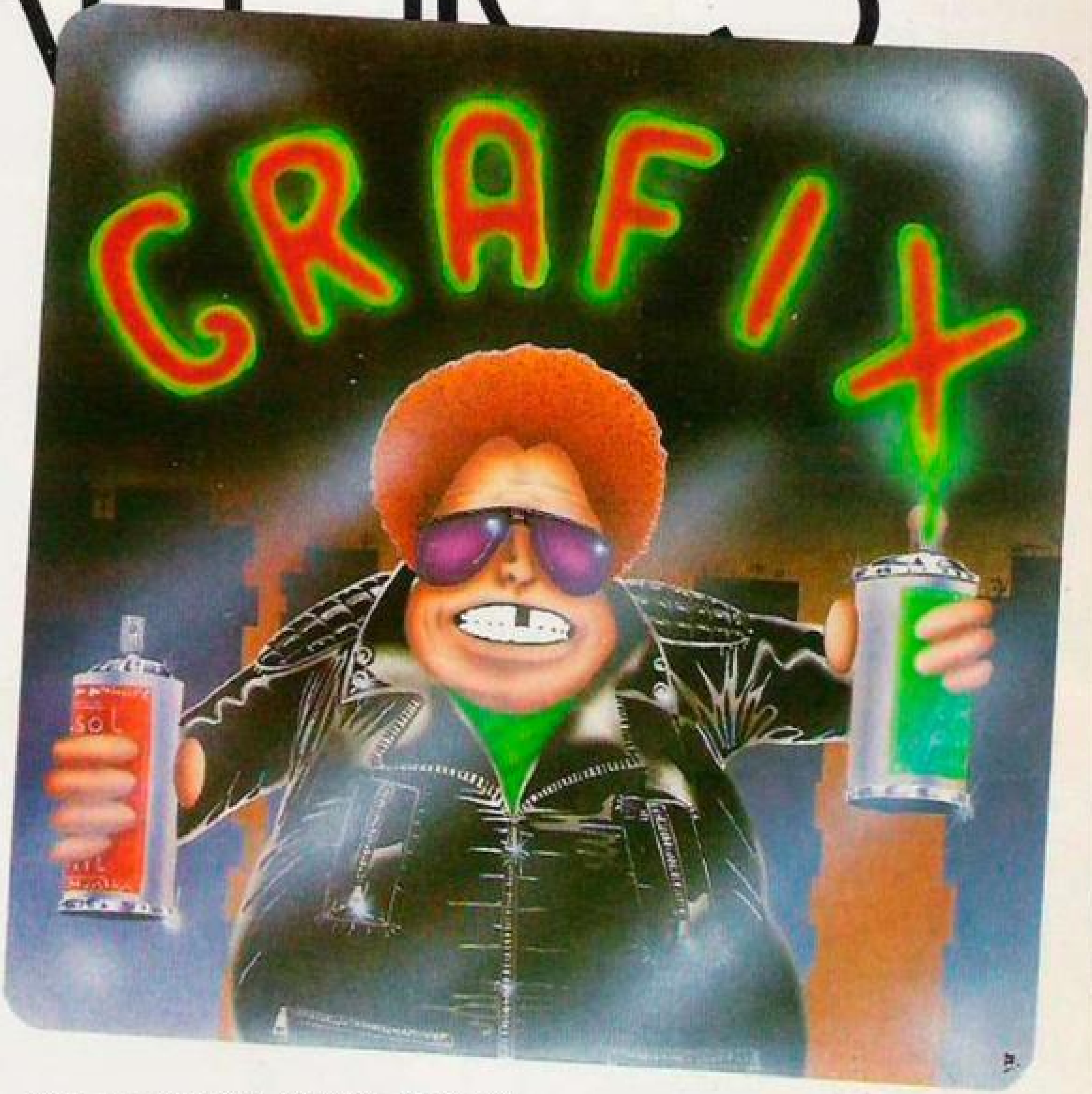
A new VDU command has been provided to colour the Mode 7 screen until it is next coloured — this command is VDU23,2 <text start line>, <text finish line>, <teletext foreground colour>, <teletext background colour>, <graphics start line>, <graphics finish line>, <teletext graphics foreground colour>, <teletext graphics background colour>.

The command colours the screen in two sections: a text section from the text start line to the text finish line <in the teletext foreground and background colours passed to it, and a graphics section in the graphics foreground and background colours sent to it.

The text or graphics areas may be as little as one line or may cover the whole screen. The start and finish lines are vertical screen "y" co-ordinates — as used by Tab(X,Y) — from the origin at the top-left of the screen.

If the text start line is set to 255 then the text parameters will be ignored and only the graphics background area will be coloured — this allows the screen to be used for graphics

GRAPHICS



only. Similarly if the graphics start line is set to 255, the graphics parameters will be ignored and only the text area will be coloured by the text commands.

The text colours are the usual teletext alpha colours, i.e., 129 to 134 — see user guide page 151 — and the graphics colours are determined by the usual teletext graphics colour code numbers in the range 145 to 151 — user guide page 155.

If the command sets text foreground colour to 0, the routine will set foreground to white. If the background colour is set to 0 by the command the routine will set the background to black. This also applies to the graphics foreground and background colours. For examples of the use of the command see table 1.

To enable printer graphics dumps and area fill algorithms I have also supplied a Point(X,Y) type command. It works in all modes, as do the other commands and returns 0 if the pixel is off, 1 if the pixel is on, and −1 — i.e., 255 — if the pixel is off the screen. The command is:

```
X% = <X co-ordinate> : Y% = <y co-
          ordinate> :
   COLOUR = (USR(Point) AND 255)
```

The state of the pixel will be returned in the variable Colour. The resident integer variables X% and Y% must contain the X and Y co-ordinates respectively of the pixel to be examined before the USR call is made. The new Plot commands may be accessed from machine code in the normal way by sending the equivalent bytes to OSWRCH of the VDU25,69,xlo,xhi,ylo,yhi command.

The Point routine may also be accessed from machine code. The assembly language given in table 2 returns the state of the pixel whose co-ordinates are stored at PX and PX+1, PY in the accumulator. To use the new Plot commands or the point subroutine the machine code in listing 1 must be merged with your own code — user guide page 402.

To start you off with graphics in these modes I have provided a set of procedures to use in all of them. Merge them into your own programs as needed:

- PROCDRAW(X1,Y1,X2,Y2) draws a line from co-ordinate (X1,Y1) to (X2,Y2).
- PROCBOX(X,Y,L1,L2) draws a box with its left-hand bottom corner at (X,Y) with a width of L1 pixels and height of L2 pixels.
- PROCELIPSE(X,Y,L,INC,MF1,MF2) plots a circle or elipse from (X,Y) with a radius of L and, for elipses, tilted INC degrees clockwise. MF1 is a modifying factor in the X direction and MF2 for the Y direction. This allows elipses to be stretched or squashed. MF factors of 1 allow true circles in modes with square pixels, e.g., Mode 6. Factors less than one will squash circles and factors greater than 1 will stretch circles.
- PROCPAINT (X,Y) flood-fills a shape, with the currently selected graphics colour, to a boundary. It commences filling from the point (X,Y).
- PROCDUMP is a printer graphics dump for the Seikosha GP-100A printer.

For other printers your favourite Mode 0 and 4 dumps should work in Modes 3 and 6 with little modification. The principal difference is that the screen resolution is equal to the actual number of screen pixels. Therefore the number of pixels stepped in the X and Y directions should be 1. The actual number of pixels in the Y direction is only 199, not 256 as in Modes 4 and 0. Any Point (X,Y) statements should be converted to their equivalent USR call given.

The routines can easily be extended by assembly language programmers, so here is how they work. The Plot commands work by intercepting the VDU vector — advanced user guide, page 261. On entry to the vector the carry flag is set if a VDU23,n, command occurred with n in the range 2 to 31.

If it did and n is equal to 2, control passes to vdu23 — my routine for colouring the Mode 7 screen.

This routine retrieves the parameters passed in the VDU call by using a Tube-compatible Peek and deposits them in the array vdu where it can manipulate them. It then clears the Mode 7 screen, prints the teletext codes down the left-hand side of the screen and sets a text window to protect them.

If carry is not set a Plot command has occurred in a non-graphics mode and the accumulator will contain the plot number. If this number is 69 my routine is called else an error message is issued.

The plot routine first sets itself up according to whether it is in Mode 3, 6 or 7 and then examines the GCol colour by Peeking. It then sets its colour masks accordingly. The byte arrays Dimmed at the start of the program are multiplication and division tables to make plotting faster. The tables are mode specific and are therefore selected each time a new command is called.

If the routine is in Mode 7 it goes to the Mode 7 plot otherwise it goes to the mode 3/6 plot routine. The routine then calculates the address of the pixel to be set or cleared — depending on the current GCol — and ORs this with those currently set.

The point routine also sets itself up depending on the current mode and immediately transfers the contents of X/ and Y% to its own variables X1 and Y1. It then calls the routine it shares with the Plot commands to calculate the address of the pixel to be examined.

The accumulator is then loaded with a value depending on the state of the pixel returned

(continued on next page)

(continued from previous page)

by this routine before returning to Basic. You should note that the Mode 7 point routine will return the ascii value of the character at the co-ordinate if it is not a graphics character. This enables the user to avoid Plotting over control codes.

Finally, the demonstration program, listing 1, and a few cautions. Once Run the program repeats a graphics demonstration using most of the graphics procedures in mode 3 then 6 and then 7. Hitting any key, after the demonstration has finished in a particular mode, will send it on to the next mode.

The code assembles from &900 onwards — this may be inconvenient as no user-defined characters and no function keys may be defined. The code is relocatable and may be assembled at any address. It may alernatively be Dimmed as indicated on line 3060.

■

```
   10MODE7
   15REM MODE3, 6, 7 GRAPHICS
   20REM (c) 1984 Chris. Bowerman.
   30DIMQX(200),QY(200)
   40PROCCODE
   50MODE3:VDU23;8964;0;0;0;23;7687;0;0;0;23;1081;0;0;
0;19,0,1;0;19,1,7;0;:PROCdemonstration(639,199)
   60A%=GET
   70MODE6:VDU23;8964;0;0;0;23;7687;0;0;0;23;1081;0;0;
0;19,0,1;0;19,1,7;0;:PROCdemonstration(319,199)
   80A%=GET
   90MODE7:VDU23,2,255,0,0,0,0,24,&91,&97:*FX1,1
  100PROCdemonstration(71,74)
  110A%=GET
  120END
  130DEFPROCdemonstration(xmax,ymax)
  140FORA%=0TO100:PLOT69,RND(xmax),RND(ymax):NEXT
  150PROCBOX(0,0,xmax,ymax)
  160PROCELIPSE(xmax/3,ymax/2,xmax/10,0,1,1):PROCELIPS
E(xmax/3,ymax/2,xmax/10,0,2,1):PROCELIPSE(xmax/3,ymax/
2,xmax/10,0,1,2)
  170PROCBOX((xmax/3)*2,ymax/4,xmax/6,ymax/4)
  180PROCELIPSE((xmax/3)*2.5,ymax/1.275,xmax/10,0,1,1)
  190PROCPAINT((xmax/3)*2.5+1,ymax/1.275+1)
  200ENDPROC
  210DEFPROCCODE
  220DIM TABLE40 49,TABLE80 49,parameter 4,dot 7,TABLE
7 74,mod 74,vdu 7
  230!dot=&08040201:dot!4=&4010
  240FORA%=0 TO 49:TABLE40?A%=0:TABLE80?A%=0:NEXT
  250C=-1:FORA%=0 TO 49 STEP2:C=C+1:val=C*40:TABLE40?A%=
val MOD256:TABLE40?(A%+1)=val DIV256:NEXT
  260C=-1:FORA%=0 TO 49 STEP2:C=C+1:val=C*80:TABLE80?A%=
val MOD256:TABLE80?(A%+1)=val DIV256:NEXT
  270C=25:FORA%=0 TO 74 STEP3:C=C-1:TABLE7?A%=C:TABLE7?(
A%+1)=C:TABLE7?(A%+2)=C:NEXT
  280C=-1:FORA%=0 TO 74:mod?A%=A% MOD3:NEXT
  290hi=&70:table=&71:save=&73:value=&74:work=&76:s
creen=&78:pixel=&7A
  300byte=&7B:temp=&7C:mask=&7D:x=&460:y=&464:X1=&7E:Y
1=&80:tube=&81:char=&75
  310rem=&76:mul=&77:index=&78:yp=&7A:xp=&7B:chr=&72:x
s=&73:ys=&74:count=&71:white=&72
  320?tube=255
  330FORscreen%=-1 TO 1:IF screen% < 0 THEN O%=0:L%=&0
900 ELSE O%=screen%*2
  340P%=L%:[OPT O%
  350BCCjump1
  360CMP#2:BNEend:JMPvdu23
  370.end
  380RTS
  390.jump1 CMP#69:BEQgo_on
  400JMPerror
  410.go_on
  420LDA#1A0:LDX#&20:JSR&FFF4
  430STX X1:STYX1+1
  440LDA#1A0:LDX#&22:JSR&FFF4
  450STX Y1
  460LDA#135:JSR&FFF4
  470CPY#7:BNEjumpon:JMPplot7
  480.jumpon CPY#3
  490BNEjump2
  500JSReighty
  510JSRstart:JMPrest
  520.jump2 CPY#6
  530BNEjump3
  540JSRfourty
  550JSRstart:JMPrest
  560.jump3
  570RTS
  580.start LDA#&F1:LDX#0:LDY#255:JSR&FFF4
  590CPX#0:BEQcolourzero:LDA#&5:STAcarryon:LDA#160:STA
carryon+3:LDA#0:STAconceal+1:JMPafter
  600.colourzero LDA#&25:STAcarryon:LDA#120:STAcarryon
+3:LDA#255:STAconceal+1
  610.after LDA#160:LDX#&39:JSR&FFF4
  620TXA:CMP#255:BNEyes:JMPno
  630.yes LDA#&25:STAhide
  640LDA#0:STAmask
  650LDA#5:STAmode
  660LDA#160:LDX#&58:JSR&FFF4
  670TXA:CMP#2:BEQjump4
  680CMP#0:BEQjump4
  690CMP#4:BNEjump5
  700LDA#&25:STAhide
  710LDA#&FF:STAmask
  720LDA#&45:STAmode
  730RTS
  740.jump4 LDA#&45:STAhide
  750LDA#&FF:STAmask
  760LDA#&25:STAmode
  770.jump5 RTS
  780.no LDA#&25:STAhide
  790LDA#&FF:STAmask
  800LDA#5:STAmode
  810LDA#160:LDX#&58:JSR&FFF4
  820TXA:CMP#2:BNEjump6
  830LDA#&25:STAmode
  840RTS
  850.jump6 CMP#3:BEQjump7
  860CMP#4:BNElastbit
  870.jump7 LDA#&45:STAmode
  880.lastbit RTS
  890.rest JSRcalculate
  900LDAscreen:STAparameter
  910LDAscreen+1:STAparameter+1
  920LDApixel
  930.hide EORmask
  940.mode ORAbyte
  950JSRpoke
  960RTS
  970.calculate LDA#199:SEC:SBCY1:STAsave
  980LDAX1:LSRA:LSRA:LSRA:STAvalue
  990LDAX1+1:AND#&7
 1000ASLA:ASLA:ASLA:ASLA:ASLA
 1010ORAvalue:STAvalue
 1020LDAX1+1
 1030LSRA:LSRA:LSRA
 1040STAvalue+1
 1050LDAsave
 1060LSRA:LSRA:LSRA
 1070STAwork
 1080ASLA
 1090STAwork
 1100TAY
 1110LDA(table),Y
 1120STAwork
 1130INY
 1140LDA(table),Y
 1150STAwork+1
 1160LDAvalue
 1170CLC
 1180ADCwork
 1190STAscreen
 1200LDAvalue+1
 1210ADCwork+1
 1220STAscreen+1
 1230ASLA:ASLA:ASLA
 1240STAscreen+1
 1250LDAscreen
 1260LSRA:LSRA:LSRA:LSRA:LSRA
 1270ORAscreen+1
 1280STAscreen+1
 1290LDAscreen
 1300ASLA:ASLA:ASLA
 1310STAscreen
 1320LDAsave
 1330AND#7
 1340STAsave
 1350CLC
 1360ADCscreen
 1370STAscreen
 1380LDA#0
 1390ADCscreen+1
 1400STAscreen+1
 1410CLC
 1420ADChimem
 1430STAscreen+1
 1440LDAX1
 1450AND#7
 1460STApixel
 1470LDA#7
 1480SEC
 1490SBCpixel
 1500STApixel
 1510LDA#1
 1520STAtemp
 1530LDYpixel
 1540STApixel
 1550CPY#0
 1560BEQjump8
 1570.loop1 LDAtemp
 1580ASLA
 1590STAtemp
 1600DEY
 1610TYA
 1620BNEloop1
 1630LDAtemp
 1640STApixel
 1650.jump8 LDAscreen:STAparameter
 1660LDAscreen+1:STAparameter+1
 1670JSRpeek
 1680STAbyte
 1690RTS
 1700.fourty LDA#&40:STAhimem
 1710LDA#TABLE40 MOD256
 1720STAtable
 1730LDA#TABLE40 DIV256
 1740STAtable+1
 1750RTS
 1760.eighty LDA#&40:STAhimem
 1770LDA#TABLE80 MOD256
 1780STAtable
 1790LDA#TABLE80 DIV256
 1800STAtable+1
 1810RTS
 1820.peek
 1830LDAtube:STAparameter+2:STAparameter+3
 1840LDA#5
 1850LDX#parameter MOD256
 1860LDY#parameter DIV256
 1870JSR&FFF1
 1880LDAparameter+4
 1890RTS
 1900.poke
 1910STAparameter+4
 1920LDAtube:STAparameter+2:STAparameter+3
 1930LDA#6
 1940LDX#parameter MOD256
 1950LDY#parameter DIV256
 1960JSR&FFF1
 1970RTS
 1980.error BRK
 1990]
 2000?P%=45
 2010$(P%+1)="Bad PLOT"
 2020P%=P%+9
 2030[OPTO%
 2040BRK
 2050.point
 2060LDA#x MOD256:STAparameter
 2070LDA#x DIV256:STAparameter+1
 2080JSRpeek
 2090STAX1
 2100LDA#(x+1) MOD256:STAparameter
 2110LDA#(x+1) DIV256:STAparameter+1
 2120JSRpeek
 2130STAX1+1
 2140LDA#y MOD256:STAparameter
 2150LDA#y DIV256:STAparameter+1
 2160JSRpeek
 2170STAY1
 2180LDA#135:JSR&FFF4
 2190CPY#7:BNEjumpover:JMPpoint7
 2200.jumpover CPY#3
 2210BNEjump9
 2220JSReighty
 2230JMPjump11
 2240.jump9 CPY#6
 2250BNEjump10
 2260JSRfourty
 2270JMPjump11
 2280.jump10 RTS
 2290.jump11 LDAY1
 2300CMP#200
 2310BCCjump12
 2320JMPoffscreen
 2330.jump12 LDY#0
 2340LDAtable,Y
 2350CMP#40
 2360BNEeight
 2370JMPfour
 2380.eight LDAX1+1
 2390CMP#2
 2400BCCjump13
 2410LDAX1
 2420CMP#127
 2430BCSout
 2440JMPjump13
 2450.out JMPoffscreen
 2460.four LDAX1+1
 2470CMP#1
 2480BCCjump13
 2490LDAX1
 2500CMP#63
 2510BCSout
 2520JMPjump13
 2530.offscreen LDA#255
 2540RTS
 2550.jump13 JSRcalculate
 2560LDApixel
 2570ANDbyte
 2580CMP#0
 2590BNEfurther
 2600RTS
 2610.further LDA#1
 2620RTS
 2630.plot7 JSRstart:JSRcursave
 2640JSRfind:LDAchr:.conceal EOR#&0:.carryon ORAchar:O
RA#160:JSR&FFEE
 2650JSRcurload:RTS
 2660.point7 LDAX1:CMP#74:BCSoutside
 2670LDAY1:CMP#75:BCSoutside:JMPrestpoint7
 2680.outside LDA#255:RTS
 2690.restpoint7 JSRcursave
 2700JSRfind:LDAchar:CMP#33:BCCendcheck:CMP#160:BCS en
dcheck:RTS:.endcheck ANDchr:CMPchr:BNEnot:JSRcurload:L
DA#1:RTS
 2710.not JSRcurload:LDA#0:RTS
 2720.find LDAX1:LSR A:STAxp
 2730LDYY1:LDAmod,Y:STArem
 2740LDA#2:SEC:SBCrem:ASLA:STAmul
 2750LDAX1:AND#1:CLC:ADCmul:STAindex
 2760TAY:LDAdot,Y:STAchr
 2770LDYY1:LDATABLE7,Y:STAyp:LDA#31:JSR&FFEE:LDAxp:JSR
&FFEE:LDAyp:JSR&FFEE
 2780LDA#135:JSR&FFF4:STXchar:RTS
 2790RTS
 2800.cursave LDA#134:JSR&FFF4:STXxs:STYys:RTS
 2810.curload LDA#31:JSR&FFEE:LDAxs:JSR&FFEE:LDAys:JSR
&FFEE:RTS
 2820.vdu23 LDA#0:STAindex:LDA#&1C:STAcount
 2830.loop2 LDA#160:LDXcount:LDY#3:JSR&FFF4:TXA:LDYind
ex
 2840STAvdu,Y:INCindex:INCcount:LDAcount:CMP#&24:BNElo
op2
 2850LDA#26:JSR&FFEE:LDA#12:JSR&FFEE
 2860 LDAvdu:CMP#255:BNEfine:JMPgraphics
 2870.fine LDA#135:STAwhite:JSRcolourline
 2880.graphics LDAvdu+4:CMP#255:BNEok:JMPwindow
 2890.ok LDAvdu+4:STAvdu:LDAvdu+5:STAvdu+1:LDAvdu+6:ST
Avdu+2:LDAvdu+7:STAvdu+3
 2900LDA#151:STAwhite:JSRcolourline
 2910.window LDA#28:JSR&FFEE:LDA#3:JSR&FFEE:LDA#24:JSR
&FFEE:LDA#39:JSR&FFEE:LDA#0:JSR&FFEE:LDA#30:JSR&FFEE:R
TS
 2920.colourline LDYvdu:INCvdu+1
 2930.loop3 LDA#31:JSR&FFEE:LDA#0:JSR&FFEE:TYA:JSR&FFE
E
 2940LDAvdu+3:CMP#0:BNEpart1:LDAvdu+2:CMP#0:BNEpart4:L
DAwhite:STAvdu+2:.part4 JSR&FFEE:JMPpart3
 2950.part1 LDAvdu+2:CMP#0:BNEpart2:LDAwhite:STAvdu+2
 2960.part2 LDAvdu+3:JSR&FFEE:LDA#157:JSR&FFEE:LDAvdu+
2:JSR&FFEE
 2970.part3 INY:CPYvdu+1:BNEloop3
 2980RTS
 2990]:IF screen% < 0 THEN :L%=&900:REM or DIM L%(P%-L
%)
 3000NEXT:!&226=L% OR(!&226 AND&FFFF0000)
 3010ENDPROC
 3020
 3030REM*********************************
 3040REM*     GRAPHICS PROCEDURES      *
 3050REM*********************************
 3060
 3070REM LINE DRAWING ROUTINE
 3080DEFPROCDRAW(X1%,Y1%,X2%,Y2%)
 3090LOCALX%,Y%,L%,A%
 3100PLOT69,X1%,Y1%
 3110X%=X2%-X1%:Y%=Y2%-Y1%
 3120L%=SQR((X1%-X2%)^2+(Y1%-Y2%)^2):IF L%=0 THEN ENDP
ROC
 3130FORA%=1 TO L%:PLOT69,X1%+A%*X%/L%,Y1%+A%*Y%/L%:NE
XT
 3140PLOT69,X2%,Y2%:ENDPROC
 3150
 3160REM SEIKOSHA GP100A PRINTER DUMP
 3170DEFPROCPDUMP(mode)
 3180IF mode=3 THEN along=643 ELSE IF mode=6 THEN alon
g=321 ELSE VDU7:ENDPROC
 3190VDU2,1,0
 3200FORXX%=6 TO 321 STEP7:REM 643 IN MO.3
 3210FORY%=0 TO 199
 3220byte=0
 3230FORx%=0 TO 6
 3240byte=byte*2
 3250X%=XX%-x%:IF (USR(point)AND255)=1 THEN byte=byte+1
 3260NEXT
 3270VDU1,byte+128
 3280NEXT
 3290VDU1,10
 3300NEXT
 3310VDU1,15,3
 3320ENDPROC
 3330
 3340REM BOX DRAWING ALGORITHM
 3350DEFPROCBOX(X%,Y%,L1%,L2%)
 3360FORA%=X% TO X%+L1%-1:PLOT69,A%,Y%:PLOT69,A%,Y%+L2
%:NEXT
 3370FORA%=Y% TO Y%+L2%:PLOT69,X%,A%:PLOT69,X%+L1%,A%:
NEXT
 3380ENDPROC
 3390
 3400REM CIRCLE DRAWING ALGORITHM
 3410DEFPROCELIPSE(X%,Y%,L,INC,MF1,MF2)
 3420A%=L/MF1:B%=L/MF2:N%=100
 3430P=2*PI/(N%-1)
 3440C1=COS(INC):S1=SIN(INC)
 3450C2=COS(P):S2=SIN(P)
 3460C3=1:S3=0
 3470x=(A%*C1):y=(A%*S1)-1
 3480FORM%=1 TO N%
 3490X1=A%*C3:Y1=B%*S3
 3500PROCDRAW(x+X%,y+Y%,(X1*C1-Y1*S1)+X%,(X1*S1+Y1*C1)
+Y%)
 3510T1=C3*C2-S3*S2
 3520S3=S3*C2+C3*S2
 3530C3=T1
 3540x=(X1*C1-Y1*S1):y=(X1*S1+Y1*C1)
 3550NEXT
 3560ENDPROC
 3570DIMQX(N),QY(N)
 3580DEFPROCSQ(X,Y,L)
 3590MOVEX,Y:DRAWX+L,Y:DRAWX+L,Y+L:DRAWX,Y+L:DRAWX,Y:E
NDPROC
 3600DEFPROCC(R,XC,YC)
 3610LOCALT
 3620MOVEXC+R,YC
 3630FORT=10TO360STEP10
 3640DRAWXC+R*COS(RAD(T)),YC+R*SIN(RAD(T))
 3650NEXT
 3660ENDPROC
 3670
 3680REM AREA FILL ROUTINE
 3690DEFPROCPAINT(SX,SY)
 3700F=1:L=0:PROCF(SX,SY)
 3710REPEAT:X=QX(F):Y=QY(F):F=(F+1) MOD200
 3720PROCF(X,Y+1):PROCF(X,Y-1):PROCF(X+1,Y):PROCF(X-1,
Y)
 3730UNTILF=(L+1)MOD200:ENDPROC
 3740DEFPROCF(X,Y):X%=X:Y%=Y:IF (USR(point) AND255) >
0 THEN ENDPROC
 3750PLOT69,X,Y:PROCQ(X,Y):ENDPROC
 3760DEFPROCQ(X,Y):L=(L+1) MOD200:QX(L)=X:QY(L)=Y:ENDPROC
```



Four great new add-ons from Ram, Number One in everything for the Spectrum…

TURBO INTERFACE

Outperforms any other Spectrum interface. Supports Kempston, Protek, Interface 2 and ROM cartridge software, with a full expansion bus at the rear. The Turbo works with two standard joysticks, making the most of the Quickshot II's rapid-fire action. Features a unique built-in power safety device. Amazing value at £22.95.

QUICKSHOT II AUTO RAPID-FIRE JOYSTICK

The top games joystick with its sleek trigger fire button and an auto-fire switch for continuous shooting – a snip at only £9.95.

SPECTRUM INTERFACE

This Kempston-compatible interface adds real games power to your Spectrum. Costs just £9.95.

SPECTRUM UPGRADE KIT

Boost your 16K Spectrum to a full 48K, allowing you to run all the latest and greatest software. Only £21.95

Where can you get your hands on Ram's amazing Spectrum add-ons? You can see them at branches of Boots, Menzies, Greens, Spectrum Group computer centres, and good computer stores everywhere.

Ram Electronics (Fleet) Ltd (Dept YC), 106 Fleet Road, Fleet, Hampshire GU13 8PA.
Credit Card hot line: *02514 25252. (Access & Visa).*

Please send me ____ Spectrum Turbo Interface(s) at £22.95.
____ Quickshot II Joystick(s) at £9.95.
____ Spectrum Interface(s) at £9.95.
____ Spectrum Upgrade Kit(s) at £21.95.
(Please state issue 2 ☐ or 3 ☐)

+£1 per order P+P (£3 Overseas) **TOTAL £**

☐ I enclose cheque/postal order ☐ charge my Access/Visa

Expiry Date___/___/___

Name________________

Address________________

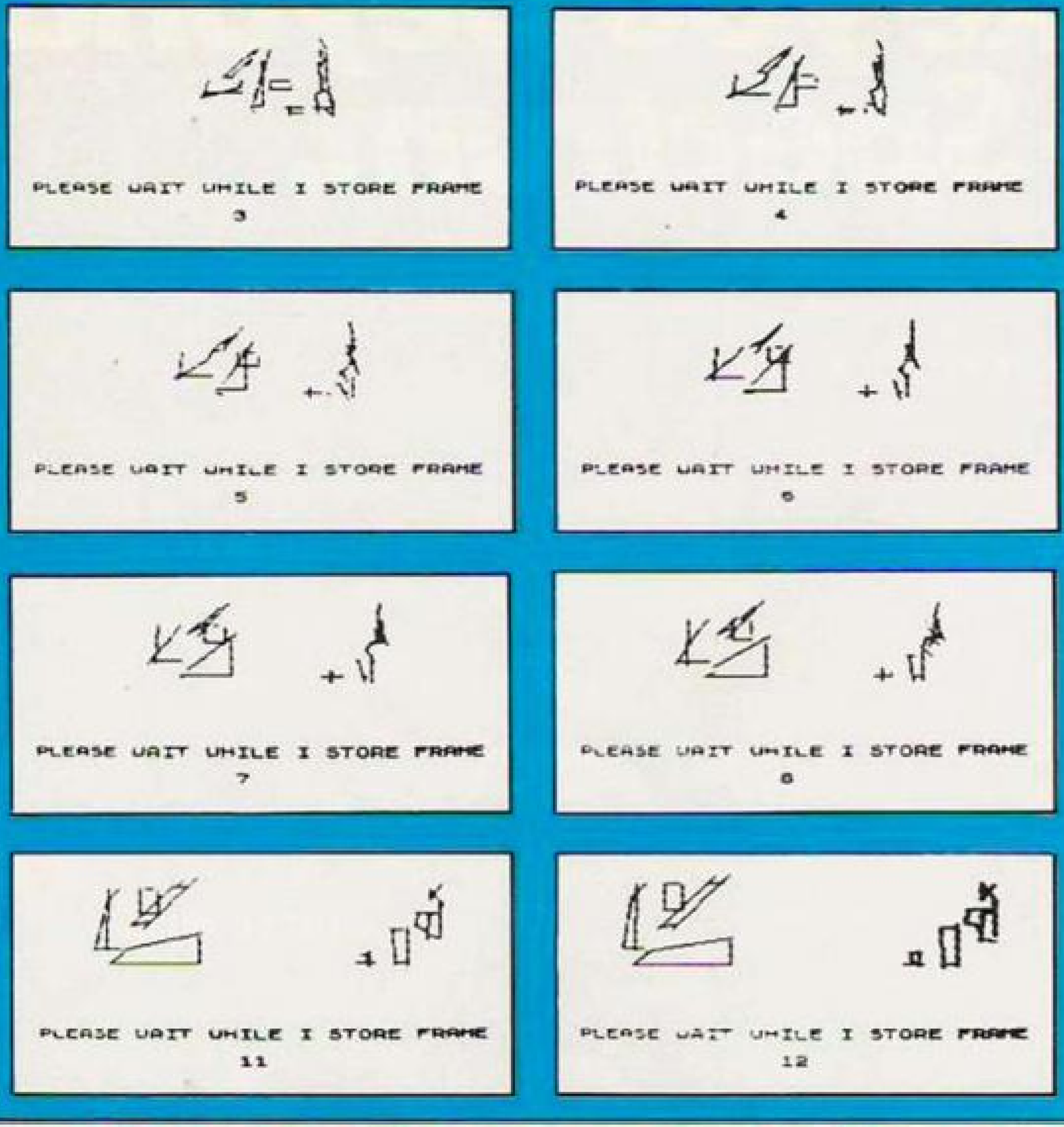
________________Postcode________________

Telephone________________

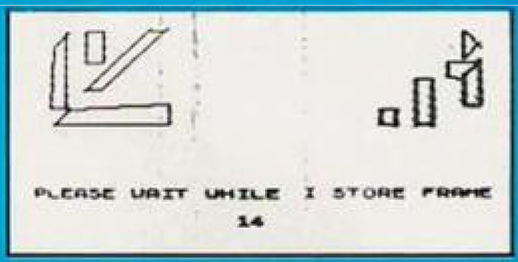
24 Hour despatch for credit cards and postal orders (7 days for cheques.)
All products are fully compatible with the Spectrum, Spectrum + and Sinclair Microdrives.

Ram Electronics (Fleet) Ltd
(Dept *YC*), 106 Fleet Road,
Fleet, Hampshire GU13 8PA.

Trade and Export enquiries welcome.







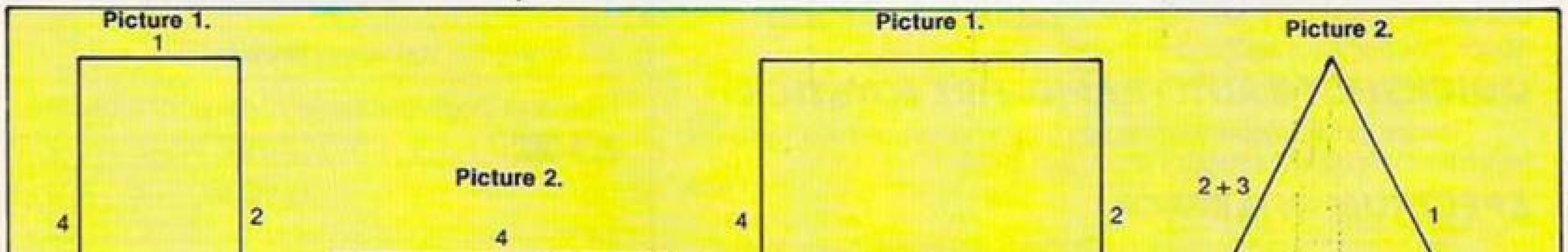
Not quite up to the standard of Channel 4's animation but getting there. Timothy Closs keeps things moving.

PIKCHACHANJA FOR YOUR SPECTRUM

Available on

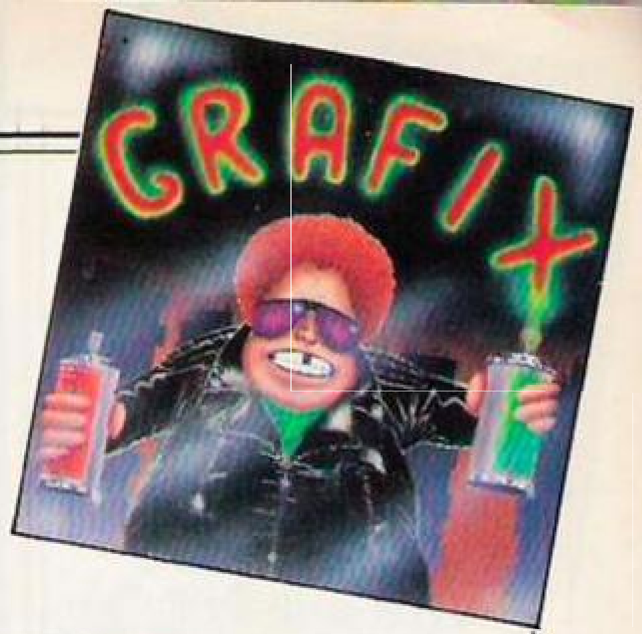


Telsoft.



Example 1. If the two pictures do not have equal numbers of lines, superimposing two or more lines will complete the picture without changing its appearance.

Example 2.




```
   4 GO SUB 3500
   5 POKE 23658,8: PAPER 7: INK
0: BORDER 7: BRIGHT 0: FLASH 0:
CLS
   6 PRINT "PRESS S TO SAVE PRES
ENT DATA": PRINT : PRINT "
L TO LOAD DATA": PRINT : PRINT "
      A TO SEE ANIMATION": PRINT
 : PRINT "        ANY OTHER KEY TO
 CONTINUE"
   7 PAUSE 0: IF INKEY$="L" THEN
 GO TO 3400
   8 IF INKEY$="S" THEN GO TO 33
00
   9 IF INKEY$="A" THEN GO TO 30
00
  10 CLS : LET KO=0: INPUT "NUMB
ER OF LINES ? ";N
  11 DIM D(N): DIM E(N): DIM Z(N
): DIM V(N)
  12 POKE 31000+N*8+1,N
  17 LET Z=150: LET X=150
  18 LET NN=10
  20 FOR G=0 TO N-1
  25 LET TI=1: GO SUB 2000: POKE
 31000+G*4,Z: POKE 31000+G*4+1,X
: LET TI=2: GO SUB 2000: POKE 31
000+G*4+2,Z: POKE 31000+G*4+3,X
  30 NEXT G
  31 LET HU=USR 32024: LET HU=US
R 32036
  32 FOR G=0 TO N-1
  35 LET TI=1: GO SUB 2000: POKE
 31000+N*4+G*4,Z: POKE 31000+N*4
+G*4+1,X: LET TI=2: GO SUB 2000:
 POKE 31000+N*4+G*4+2,Z: POKE 31
000+N*4+G*4+3,X
  40 NEXT G
  41 FOR G=10 TO 20: BEEP .05,G:
 NEXT G
  42 CLS
  50 FOR G=0 TO N-1
  60 LET D(G+1)=(PEEK (31000+N*4
+G*4)-PEEK (31000+G*4))/14
  70 LET E(G+1)=(PEEK (31000+N*4
+G*4+1)-PEEK (31000+G*4+1))/14
  72 LET Z(G+1)=(PEEK (31000+N*4
+G*4+2)-PEEK (31000+G*4+2))/14
  75 LET V(G+1)=(PEEK (31000+N*4
+G*4+3)-PEEK (31000+G*4+3))/14
  76 NEXT G
  80 DIM C(N): DIM U(N): DIM S(N
): DIM N(N)
  90 FOR G=0 TO N-1: LET C(G+1)=
PEEK (31000+G*4): LET U(G+1)=PEE
K (31000+G*4+1): LET S(G+1)=PEEK
 (31000+G*4+2): LET N(G+1)=PEEK
 (31000+G*4+3)
 100 NEXT G
 105 LET NUM=120
 110 FOR M=1 TO 15
 115 PRINT AT 12,0;"PLEASE WAIT
WHILE I STORE FRAME ";AT 14,14;M
 120 FOR G=1 TO N
 130 PLOT C(G),U(G): DRAW S(G)-C
(G),N(G)-U(G)
 140 NEXT G
 150 FOR G=1 TO N
 160 LET C(G)=C(G)+D(G)
 170 LET U(G)=U(G)+E(G)
 180 LET S(G)=S(G)+Z(G)
 190 LET N(G)=N(G)+V(G)
 200 NEXT G
 205 POKE 32002,NUM: LET A=USR 3
2000
 206 LET NUM=NUM+8
 210 LET HU=USR 32036: NEXT M
 211 CLS
 215 GO TO 3000
 220 GO TO 50
1000 FOR G=1 TO N: PRINT G;" ";:
 PRINT X(G),Y(G),A(G),B(G)
1010 NEXT G
1020 STOP
2000 IF INKEY$="5" AND Z>NN-1 TH
EN LET Z=Z-NN
2005 IF KO=10 THEN GO TO 3210
2010 IF INKEY$="6" AND X>111+NN
THEN LET X=X-NN
2020 IF INKEY$="7" AND X<176-NN
THEN LET X=X+NN
2030 IF INKEY$="8" AND Z<256-NN
THEN LET Z=Z+NN
2035 IF INKEY$="1" THEN LET NN=1
2036 IF INKEY$="2" THEN LET NN=1
0
2040 IF INKEY$="0" THEN BEEP .2,
20: GO TO 3230
2045 IF INKEY$="9" THEN BEEP .2,
25: PLOT Z,X: DRAW LZ-Z,LX-X: GO
 TO 3200
2050 PLOT Z,X
2052 PRINT AT 10,0;"X COORD - ";
Z;" ";AT 10,16;"Y COORD - ";X;"
";AT 12,8;"JUMP VALUE - ";NN;" "
2053 PRINT AT 14,8;"LINE NUMBER
- ";G;"  ": PAUSE 0
2055 OVER 1: PLOT Z,X: OVER 0
2060 GO TO 2000
3000 FOR H=1 TO 20: NEXT H: FOR
G=1 TO 15
3010 POKE 32017,G*8+120: LET A=U
SR 32012
3011 IF INKEY$<>"" THEN GO TO 5
3015 PAUSE 2
3020 NEXT G
3030 PAUSE 20
3040 FOR G=15 TO 1 STEP -1
3050 POKE 32017,G*8+120: LET A=U
SR 32012
3051 IF INKEY$<>"" THEN GO TO 5
3055 PAUSE 2
3060 NEXT G
3070 PAUSE 5
3080 GO TO 3000
3200 LET NZ=Z: LET NX=X: LET Z=L
Z: LET X=LX: LET KO=10: RETURN
3210 LET Z=NZ: LET X=NX: LET LZ=
Z: LET LX=X: LET KO=0: RETURN
3230 IF TI=2 THEN PLOT Z,X: DRAW
 LZ-Z,LX-X
3240 LET LZ=Z: LET LX=X: RETURN
3300 CLS : INPUT "NAME OF FILE ?
 ";N$: IF LEN (N$)>10 THEN GO TO
 3300
3310 POKE 30999,N: SAVE N$CODE 3
0999,N*8+1
3320 VERIFY N$CODE
3330 GO TO 5
3400 CLS : INPUT "FILE NAME ? ";
N$: IF LEN (N$)>10 THEN GO TO 34
00
3410 LOAD N$CODE
3411 LET N=PEEK (30999): DIM D(N
): DIM E(N): DIM Z(N): DIM V(N)
3420 GO TO 41
3500 FOR G=32000 TO 32049: READ
A: POKE G,A: NEXT G
3505 RESTORE
3510 RETURN
3520 DATA 17,0,240,33,0,64,1,0,8
,237,176,201,17,0,64,33,0,208,1,
0,8,237,176,201,17,0,80,33,0,64,
1,0,8,237,176,201,33,0,64,54,0,1
7,1,64,1,255,7,237,176,201
```

THANKS TO A technique like the one American animation experts use, Pikchachanja allows one picture to be transformed into another in a smooth, 15-frame animation sequence.

The pictures consist of a series of straight lines, with each line having a partner in the other picture. By changing each line into its partner you can achieve up to 125 lines, and by paring lines differently the animation can be changed, so prior planning can add a lot to the finished product.

When you've finished typing in the program, run it and you'll see the main menu:

PRESS S TO SAVE PRESENT DATA
L TO LOAD DATA
A TO SEE ANIMATION
ANY OTHER KEY TO CONTINUE

Press any key other than S, L and A and the computer will ask how many lines are required in the pictures — it must be the same number for each picture as every line must have one partner, but see note with example 1.

The screen will then clear and the cursor will appear, together with information concerning the cursor position, the number of lines entered and the present cursor jump in pixels.

The pictures are confined to the upper third of the screen for memory reasons. Now you can enter the first picture lines in one of two ways:

1. Plot a point, move cursor and plot another point. A line is drawn from one point to the other.
2. Draw a line from the last point plotted on the previous line to the cursor.

Apart from using the cursor keys to move the cursor, here are the other controls:

- 1 sets cursor jump to one pixel
- 2 sets cursor jump to 10 pixels
- 0 plots a point
- 9 draws a line from last point plotted

When all the lines have been entered, a short beep transfers the first picture to the lower third of the screen. Now you can draw the second picture in the same way, entering lines in the appropriate order so that pairing is automatic.

The Spectrum will then proceed to draw up each intermediate frame on the screen and store them in its memory. When all 15 frames have been stored, the screen will clear and the pictures are transformed back and forth in a smooth sequence, during which the main menu can be recalled by the touch of a button.

The Spectrum will then proceed to draw up each intermediate frame on the screen and store them in its memory. When all 15 frames have been stored, the screen will clear and the pictures are transformed back and forth in a smooth sequence, during which the main menu can be recalled by the touch of a button.

The data held in the computer can be saved by pressing S while the main menu is displayed.

Data can be loaded by pressing L while the menu is displayed. Having entered the file name the Spectrum will search for the file and, if found, will begin to draw up the frames and store them in its memory as mentioned earlier. The animation sequence present in memory can be viewed by pressing A.

For a copy of Pikchachanja together with many other novel applications, contact me at 5 Western Drive, Shepperton, Middlesex, TW17 8HJ. Simply send a blank tape and an SAE together with 99p for recorded data which can be loaded directly into your Spectrum. ■



Channel 4 spent £50,000 to get that pixel-spinning logo. Nalin Sharma adapts Tim Closs' program for an economy version for your CBM-64.

SO IMPRESSED were we by Timothy Closs' Pikchachanja program for the Spectrum — on page 94 — that we asked Nalin Sharma to convert it for the CBM-64. His version works in the same way as the Spectrum program except that it offers 18 rather than 15 frames

CBM-64 PIKCHA

Listing 1.

```
5 s=7497:en=8817:sm=0:chk=69229:q$=chr$(34
)
10 forx=stoen:reada:pokex,a
20 sm=sm+a:next
30 ifsm<>chkthenprint"data error":end
100 d$=",1":rem***d$=",8" for disk ***
110 print"▌    "q$"cart"q$d$
120 poke631,19:poke632,131:poke198,2
130 end
10000 data 165, 248, 74, 74, 41, 254
10010 data 168, 185, 236, 30, 133, 252
10020 data 185, 235, 30, 133, 251, 165
10030 data 248, 41, 7, 168, 165, 250
10040 data 41, 7, 170, 165, 250, 41
10050 data 248, 24, 101, 251, 133, 251
10060 data 165, 252, 101, 249, 133, 252
10070 data 96, 32, 73, 29, 189, 29
10080 data 31, 17, 251, 145, 251, 96
10090 data 32, 73, 29, 189, 37, 31
10100 data 49, 251, 145, 251, 96, 169
10110 data 0, 133, 249, 133, 251, 169
10120 data 32, 133, 252, 165, 250, 10
10130 data 10, 10, 24, 105, 64, 133
10140 data 250, 160, 0, 177, 251, 145
10150 data 24 , 169, 0, 145, 251, 230
10160 data 249, 230, 251, 208, 242, 230
10170 data 250, 230, 252, 165, 252, 201
10180 data 40, 208, 232, 96, 165, 1
10190 data 41, 254, 133, 1, 169, 0
10200 data 133, 249, 133, 251, 169, 32
10210 data 133, 252, 165, 250, 10, 10
10220 data 10, 24, 105, 64, 133, 250
10230 data 160, 0, 177, 249, 145, 251
10240 data 230, 249, 230, 251, 208, 246
10250 data 230, 250, 230, 252, 165, 252
10260 data 201, 40, 208, 236, 165, 1
10270 data 9, 1, 133, 1, 96, 169
10280 data 0, 133, 251, 169, 32, 133
10290 data 252, 169, 192, 133, 249, 169
10300 data 40, 133, 250, 160, 0, 177
10310 data 251, 145, 249, 165, 249, 24
10320 data 105, 1, 133, 249, 165, 250
10330 data 105, 0, 133, 250, 230, 251
10340 data 208, 235, 230, 252, 165, 252
10350 data 201, 40, 208, 227, 96, 32
10360 data 189, 30, 32, 162, 30, 120
10370 data 169, 81, 141, 20, 3, 169
10380 data 30, 141, 21, 3, 169, 1
10390 data 133, 247, 141, 26, 208, 173
10400 data 17, 208, 41, 127, 141, 17
10410 data 208, 169, 162, 141, 18, 208
10420 data 169, 255, 141, 25, 208, 169
10430 data 0, 141, 14, 220, 88, 96
10440 data 165, 247, 201, 2, 240, 36
10450 data 173, 17, 208, 41, 223, 141
10460 data 17, 208, 169, 21, 141, 24
10470 data 208, 230, 247, 169, 129, 141
10480 data 25, 208, 173, 17, 208, 41
10490 data 127, 141, 17, 208, 169, 49
10500 data 141, 18, 208, 76, 49, 234
10510 data 173, 17, 208, 9, 32, 141
10520 data 17, 208, 169, 29, 141, 24
10530 data 208, 198, 247, 169, 129, 141
10540 data 25, 208, 173, 17, 208, 41
10550 data 127, 141, 17, 208, 169, 162
10560 data 141, 18, 208, 104, 168, 104
10570 data 170, 104, 64, 169, 0, 133
10580 data 251, 169, 32, 133, 252, 160
10590 data 0, 169, 0, 145, 251, 230
10600 data 251, 208, 248, 230, 252, 165
10610 data 252, 201, 64, 208, 240, 96
10620 data 169, 0, 133, 251, 169, 4
10630 data 133, 252, 165, 249, 10, 10
10640 data 10, 10, 5, 248, 133, 248
10650 data 165, 248, 145, 251, 165, 251
10660 data 24, 105, 1, 133, 251, 165
10670 data 252, 105, 0, 133, 252, 201
10680 data 6, 208, 235, 165, 251, 201
10690 data 48, 208, 229, 96, 0, 32
10700 data 64, 33, 128, 34, 192, 35
10710 data 0, 37, 64, 38, 128, 39
10720 data 192, 40, 0, 42, 64, 43
10730 data 128, 44, 192, 45, 0, 47
10740 data 64, 48, 128, 49, 192, 50
10750 data 0, 52, 64, 53, 128, 54
10760 data 192, 55, 0, 57, 64, 58
10770 data 128, 59, 192, 60, 0, 62
10780 data 128, 64, 32, 16, 8, 4
10790 data 2, 1, 127, 191, 223, 239
10800 data 247, 251, 253, 254, 120, 169
10810 data 49, 141, 20, 3, 169, 234
10820 data 141, 21, 3, 88, 169, 1
10830 data 141, 14, 220, 169, 21, 141
10840 data 24, 208, 173, 17, 208, 41
10850 data 223, 141, 17, 208, 169, 0
10860 data 141, 26, 208, 96, 253
```

Listing 2.

```
10 POKE53281,15
40 POKE55,70:POKE51,70:POKE56,29
:POKE52,29:CLR
50 POKE650,128
58 MOVEDN=7664
60 LODFRAME=7689
62 PLOT=7540
64 SAVFRAME=7562
66 UNPLOT=7551
68 SETUP=7712
70 ZP=248
80 N=70
100 REM** X,Y = LAST POINT **
110 REM** X1,Y1 = NEXT POINT **
120 REM** T=TANGENT Y'/X' **
150 REM** N = NUMBER OF POINTS *
*
160 REM** DI = PIXEL INCREMENTS
**
165 REM** F = NUMBER OF FRAMES *
*
170 REM**
175 REM**
180 REM**
185 REM**
200 DIMA%(N,4)
210 FORCP=0TON:FORJ=0TO4
220 A%(CP,J)=0:NEXT:NEXT
230 X=50:X1=X:Y=25:Y1=Y:DI=5:F=1
7
240 PRINT"
      1=DRAW PICTURES"
250 POKEZP,1:POKEZP+1,0:SYS SETU
P:REM***PAPER(ZP)***INK(ZP+1)***

270 PRINT"                 2=SAVE DATA
"
280 PRINT"                 3=LOAD DATA
"
290 PRINT"                 4=SEE ANIMA
TION"
300 GETA$:IFA$=""THEN300
310 A=VAL(A$):IFA<1ORA>4THEN300
313 PRINT"":POKEZP,1:POKEZP+1,0
:SYS SETUP:REM***PAPER(ZP)***INK
(ZP+1)***
317 C=X:D=Y:GOSUB1000
320 ON A GOSUB2910,4000,5000,600
0
330 GOTO240
350 END
360 REM ***
365 REM *** DRAW 1ST PICTURE ***

368 REM ***
370 FORCP=0TON:PRINT"
"TAB(25)CP"▌  ":PRINTTAB(25
)DI"▌ □"
400 GOSUB700
405 IFA$="£"THEN 485:REM**FINISH
**
410 IFA$="▌"THEN 450:REM**LINE(F
5)**
415 REM*** HANDLE PLOT (F1) ***
420 C=X1:D=Y1:GOSUB1000
430 A%(CP,4)=0:REM**JUST PLOT**
440 GOTO480
445 REM*** HANDLE LINE (F5) ***
450 IF CP=0THEN420:REM**CHECK**
455 GOSUB800
460 A%(CP,4)=1:REM**PLOT & LINE*
*
480 X=X1:Y=Y1:A%(CP,0)=X:A%(CP,1
)=Y:NEXT
485 A%(CP,4)=2:CP=N+1:RETURN
492 REM ***
495 REM *** DRAW LAST PICTURE **
```

of animation.

When the program is run, a menu appears with four options: 1 Draw Picture; 2 Save Data; 3 Load Data; 4 See Animation. Details of these options are given below. For a more general introduction to this cartoon facility you are advised to consult Timothy Closs' article.

Controls for drawing are:

A/Z = Up/Down
Cursor up/Cursor left = Left/Right
* = Decrease Pixel Jump
↑ = Increase Pixel Jump
£ = Finish
Fl = Plot Point
F5 = Draw line from last point

Once the £ key has been pressed, you must draw the last picture using the same number of points. However, this time the £, F1 and F5 keys will have the same function — either a point will be plotted, a line will be drawn or the program will return to the menu according to how the first picture was drawn. Once completed, the program will draw and store frames 1 to 16 and will then return to the menu.

Save data — saves the contents of the array A% (CP,G) to tape or disc. Then it returns you to the menu.

Load data — loads data from tape or disc, and then draws or stores frames 0-17, followed by a return to the menu.

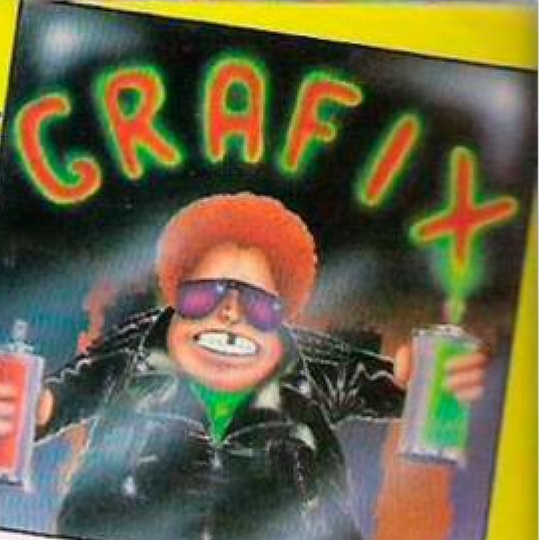
See animation — shows all 18 frames in forward and then backward sequence. This continues until Return is pressed.

To enter the program:

1. Type in listing 1 — check and alter for cassette/disc and then save "CLOADER".
2. Type in listing 2 — check and save "CART" after "CLOADER". Please note in line 80 in listing 2, N is the maximum number of points available. It can be increased from 70 by cutting out all the Rem statements. Each frame takes up 2,048 bytes. Four frames are stored under the Basic Rom.

Memory map.

$0800—$1D45 = Basic program
$1D49—$1F51 = Machine code
$2000—$3FFF = Hi-res screen
$4000—$CFFF = Storage for 18 frames



4 CHANJA

```
*
498 REM ***
500 FORCP=0TON:PRINT"SQQQQQQQQQQ
QQQQQ"TAB(25)CP"] ":PRINTTAB(25
)DI"] C"
510 IFA%(CP,4)=2THEN620
520 GOSUB700
530 IFA%(CP,4)=1THEN590
540 REM*** HANDLE PLOT (F1) ***
550 C=X1:D=Y1:GOSUB1000
570 GOTO610
580 REM*** HANDLE LINE (F5) ***
590 IFCP=0THEN550:REM**CHECK**
595 GOSUB'
610 X=X1:   _:A%(CP,2)=X:A%(CP,3
)=Y:NEXT
620 CP=N+1:RETURN
692 REM ***
696 REM *** GET KEY SUB ***
699 REM ***
700 GETA$:IFA$=""THEN700
705 C=X1:D=Y1:GOSUB2000:REM**UNP
LOT**
710 IFA$="A"THENY1=Y1-DI:GOTO774
:REM **UP**
720 IFA$="Z"THENY1=Y1+DI:GOTO776
:REM **DN**
730 IFA$="]"THENX1=X1+DI:GOTO772
:REM **RT**
740 IFA$="Q"THENX1=X1-DI:GOTO770
:REM **LF**
750 IFA$="■"ORA$="]"ORA$="£"THEN
 RETURN:REM **F1=PLOT & F5=LINE
& £=FINISH **
755 IFA$="*"THENDI=DI-1:IFDI<1TH
ENDI=1
760 IFA$="↑"THENDI=DI+1:IFDI>10T
HENDI=10
762 PRINTTAB(25)DI"] □"
765 C=X1:D=Y1:GOSUB1000:GOTO700
770 IFX1<0THENX1=319:GOTO782
772 IFX1>319THENX1=0:GOTO782
774 IFY1<0THENY1=0:GOTO782
776 IFY1>47THENY1=47:GOTO782
782 C=X1:D=Y1:GOSUB1000:REM**PLO
T**
784 GOTO700
790 REM ***
799 REM *** DRAW LINE SUB ***
800 IF X1=X THEN 865
805 IF Y1=Y THEN 900
810 T=(Y1-Y)/(X1-X):H=SQR((X1-X)
↑2+(Y1-Y)↑2)
820 S=(X1-X)/H:REM **AUTO STEP D
IFF**
830 FORI=XTOX1STEP S
840 Y3=INT(Y+(T*(I-X))+.5)
850 C=I:D=Y3:GOSUB1000
860 NEXT:RETURN
862 REM *** HORIZONTAL LINE ***
865 IF Y1=Y THEN RETURN
870 FOR Y3=YTOY1STEP SGN(Y1-Y)
880 C=X:D=Y3:GOSUB1000
890 NEXT:RETURN
895 REM *** VERTICAL LINE ***
900 FOR I=XTOX1STEP SGN(X1-X)
910 C=I:D=Y:GOSUB1000
920 NEXT:RETURN
990 REM*** PLOT SUB ***(C,D)***
1000 POKEZP,D:POKEZP+1,0:IFC<256
THEN1020
1010 POKEZP+1,1:C=C-256
1020 POKEZP+2,C:SYSPLOT:RETURN
1990 REM*** UNPLOT SUB ***(C,D)*
**
2000 POKEZP,D:POKEZP+1,0:IFC<256
THEN2020
2010 POKEZP+1,1:C=C-256
2020 POKEZP+2,C:SYSUNPLOT:RETURN

2900 REM*** DRAW PICTURES ***
2910 PRINT"SQQQQQQQQQQQQQQQQ
     POINT NUMBER="
2920 PRINT"                PIXEL JUMP
=□"
3000 GOSUB370:SYSMOVEDN
3010 POKEZP+2,0:SYSSAVFRAME
3016 C=X:D=Y:GOSUB1000
3020 GOSUB500
3030 POKEZP+2,17:SYSSAVFRAME
3090 FORW=1TO(F-1)
3100 FORCP=0TON
3110 IFA%(CP,4)=2THEN3185
3120 X1=A%(CP,0)+INT(((A%(CP,2)-
A%(CP,0))*W/17)+.5)
3130 Y1=A%(CP,1)+INT(((A%(CP,3)-
A%(CP,1))*W/17)+.5)
3140 IFA%(CP,4)=0THENC=X1:D=Y1:G
OSUB1000
3160 IFA%(CP,4)=1THENGOSUB800
3180 X=X1:Y=Y1:NEXT
3185 CP=N+1:SYSMOVEDN
3190 POKEZP+2,W:SYSSAVFRAME
3200 NEXTW:RETURN
3990 REM*** SAVE DATA  ***
4000 SYS7981
4002 PRINT"QQQQQQQQQQQQQQQQ  SAV
ING DATA "
4005 PRINT"ENTER FILE NAME"
4010 INPUTNM$
4020 PRINT"]T■APE OR ]D■ISK ?"
4030 GETX$:IFX$=""THEN4030
4040 IFX$="T"THEN4080
4050 IFX$<>"D"THEN4030
4060 OPEN1,8,4,"0:"+NM$+",S,W"
4070 GOTO4090
4080 OPEN1,1,1,NM$
4090 FORCP=0TON:FORG=0TO4
4100 BT=A%(CP,G):PRINT#1,BT
4110 NEXT
4120 NEXT:CLOSE1
4130 RETURN
4990 REM*** LOAD DATA  ***
5000 SYS7981
5002 PRINT"QQQQQQQQQQQQQQQQ  LOA
DING DATA "
5005 PRINT"ENTER FILE NAME"
5010 INPUTNM$
5020 PRINT"]T■APE OR ]D■ISK ?"
5030 GETX$:IFX$=""THEN5030
5040 IFX$="T"THEN5080
5050 IFX$<>"D"THEN5030
5060 OPEN1,8,4,"0:"+NM$+",S,R"
5070 GOTO5090
5080 OPEN1,1,0,NM$
5090 FORCP=0TON:FORG=0TO4
5100 INPUT#1,BT:A%(CP,G)=BT
5110 NEXT
5120 NEXT:CLOSE1
5125 POKEZP,1:POKEZP+1,0:SYS SET
UP
5130 FORW=0TOF
5140 FORCP=0TON
5150 IFA%(CP,4)=2THEN5210
5160 X1=A%(CP,0)+INT(((A%(CP,2)-
A%(CP,0))*W/17)+.5)
5170 Y1=A%(CP,1)+INT(((A%(CP,3)-
A%(CP,1))*W/17)+.5)
5180 IFA%(CP,4)=0THENC=X1:D=Y1:G
OSUB1000
5190 IFA%(CP,4)=1THENGOSUB800
5200 X=X1:Y=Y1:NEXT
5210 CP=N+1:SYSMOVEDN
5220 POKEZP+2,W:SYSSAVFRAME
5230 NEXTW:RETURN
5990 REM*** SEE ANIMATION ***
6000 PRINT"SQQQQQQQQQQQQQQQQ  ANI
MATION "
6005 FORW=0TOF
6010 POKEZP+2,W:SYSLODFRAME
6030 NEXT
6035 GETA$:IFA$=CHR$(13)THENRETU
RN
6040 FORW=FTO0STEP-1
6050 POKEZP+2,W:SYSLODFRAME
6070 NEXT
6100 GOTO6005
```

■

Are you brave enough to enter the savage world of mighty Conan?

You've experienced his barbaric world in movies, novels and comics. Now Conan is ready to spring to life in your home.

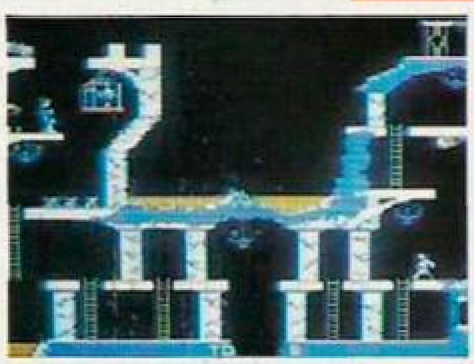
So flex your muscles and prepare to do battle.

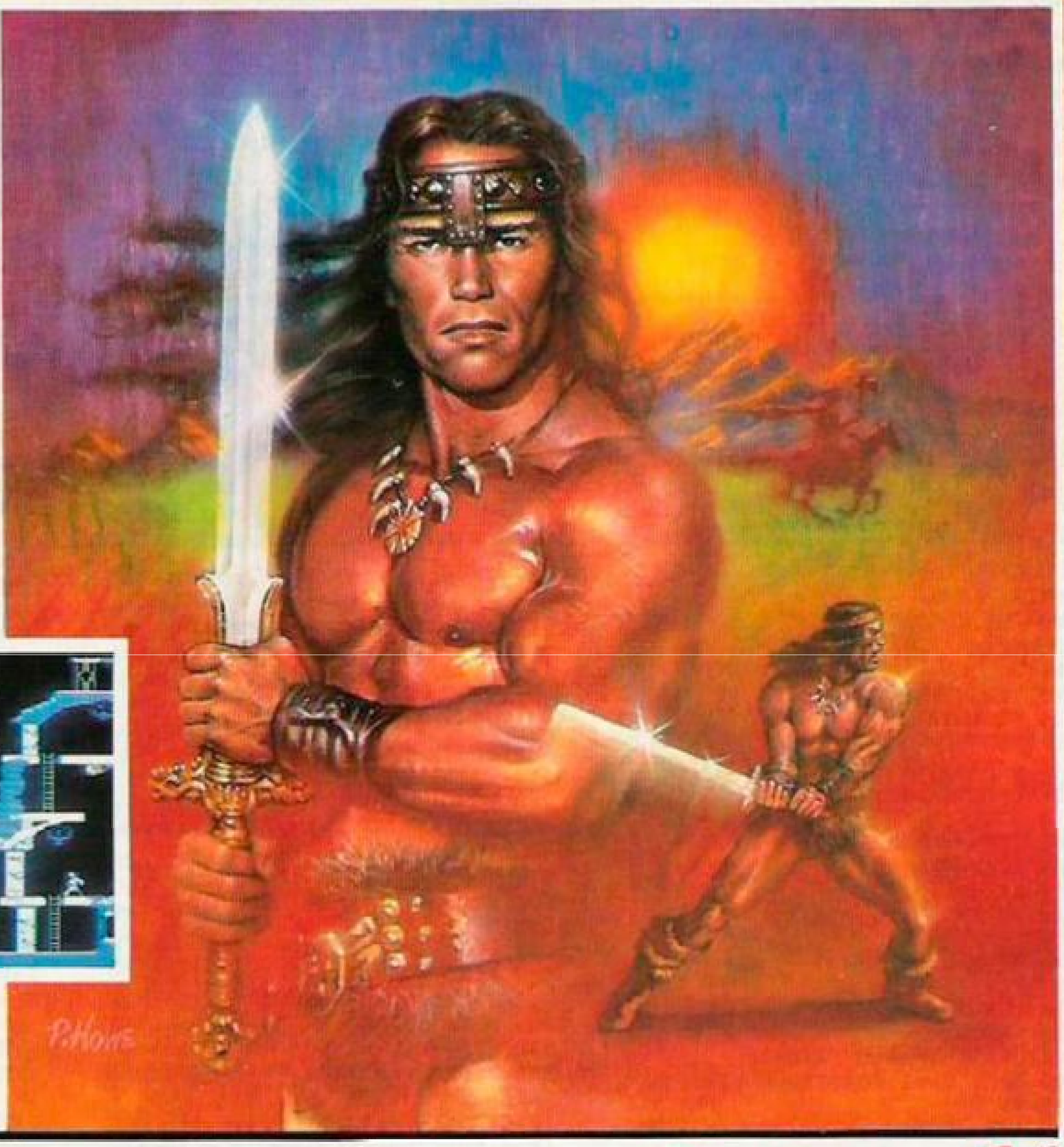
Your mission is to slay Conan's wicked rival and bring home a bounty of gems. Blocking your way are fire-breathing dragons, poisonous scorpions, vicious bats and other deadly adversaries.

If that wasn't enough, you also have to fight your way through dark caves and dangerous dungeons, while looking out for mysterious creatures waiting their moment to attack!

ATARI

64





Commodore 64/ Atari Disk	£14.95
Commodore 64/ Atari Cassette	£9.95

FAMOUS FACES™ from Datasoft®

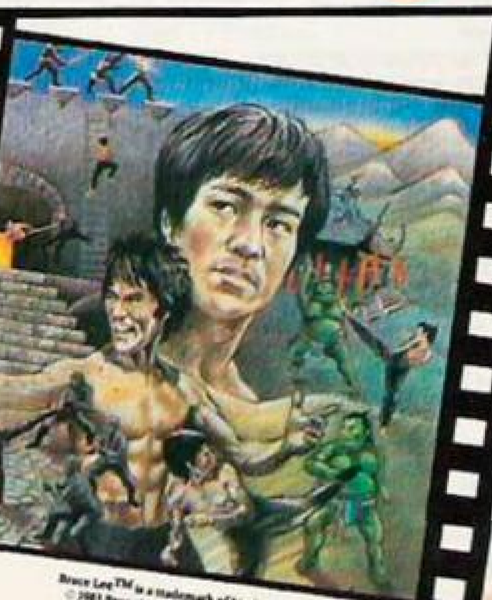
THE LEGENDARY BRUCE LEE

• Twenty Secret Chambers • Dazzling Graphics
• Unique Multiple Player Options

Spectrum 48K	£7.95
Commodore 64 Cassette	£9.95
Commodore 64 Disk	£14.95
Atari Disk/Cassette	£14.95

U.S. GOLD All American Software

U.S. Gold Limited, Unit 10,
Parkway Industrial Centre, Heneage Street,
Birmingham B7 4LY. Tel: 021-359 3020.



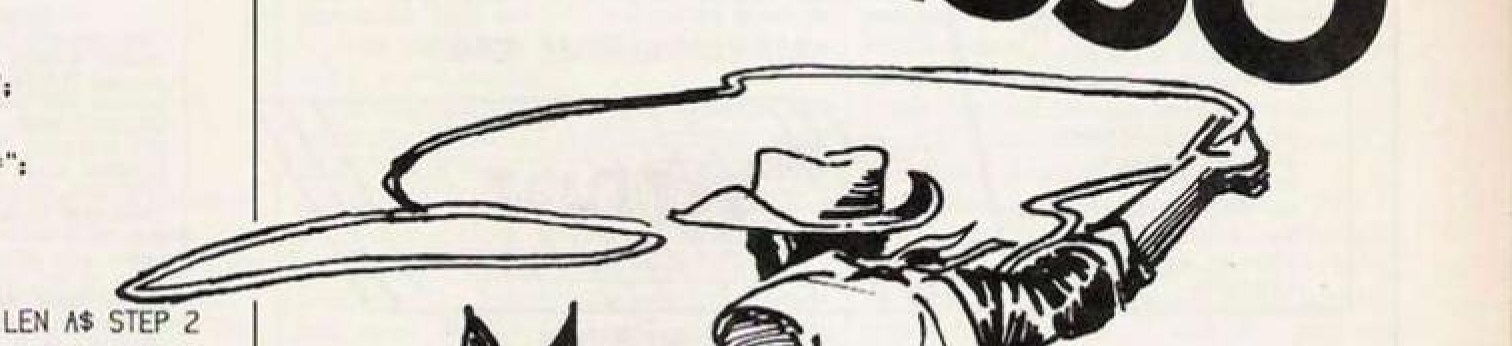
Yee-ha! Git those screen blocks movin'. Mart Stenlake cuts those screen-handling problems down to size.

ZX-81 SCREEN LASSO



Listing 1.
Hex loader.

```
10  REM HEX LOADER
20  GOSUB 240
30  FOR N=S TO F STEP 8
40  SCROLL
50  LET T=0
60  PRINT N; "-";
70  INPUT A$
80  PRINT A$; "=";
90  INPUT TOT
100 PRINT TOT
110 LET Z=0
120 FOR K=1 TO LEN A$ STEP 2
130 LET C=(CODE A$(K)-28)*16
    + CODE A$(K+1)-28
140 LET T=T+C
150 POKE N+Z,C
160 LET Z=Z+1
170 NEXT K
180 IF TOT=T
    THEN
    GOTO 220
190 SCROLL
200 PRINT "ERROR -
    PLEASE INPUT
    AGAIN"
210 GOTO 40
220 NEXT N
230 STOP
240 PRINT "START ADDRESS"
250 INPUT S
260 PRINT "FINISH ADDRESS"
270 INPUT F
280 RETURN
```



SCREEN IS A machine-code program for the 16K ZX-81, enabling the creation, editing, moving and storing of any rectangular block of the screen. The following facilities are included.

- Any block size from one character to a full screen may be defined on the screen.
- There is full on-screen editing, and all single characters — including graphics and inverse — are available directly from the keyboard.
- The whole block is stored in a Rem which is automatically adjusted in size.
- Cursor movement may be selected as either automatic increment which is useful for text input and character repeats, or cursor key dependent which will assist in the creation of graphic designs.
- Once defined, the block may be moved on screen using the cursor keys. It may be erased, printed in its last position, or the print position may be altered to any part of the screen.
- Also included is a fast screen clear.

Possible uses for Screen include creating and storing graphics pictures, animation of games, saving and printing text and data.

The program is 1,066 bytes long and is contained in two Rem statements. To enter the program, first type in 1 Rem and 535 characters. This is best done in fast mode, repeating one character until three short of a full line on the 17th line.

If correct, Print Peek 17049 should give 118. Edit this line to 2 Rem and delete four characters. Print Peek 17586 should give 118. Type in the hex loader program in listing 1. Using this, first enter the machine code in listing 2 — start 16514, finish 17048. Then run again to enter listing 3 — start 17055, finish 17585.

To save time in the event of a crash it is wise to save the program at this stage.

Testing can only be effectively carried out by following the instructions.

If a crash or problems occur, then the machine code may be checked using the program in listing 4. Hold down any key and listing will progress. The program line following 2 Rem cannot be removed at any time, but it may be edited to a minimum length or used with this program, e.g., 3 Rem.

To enable the program to run, an existing Rem statement must be specified. The Rem

(continued on page 101)

(continued from page 99)

should initially contain no characters, and any line number, except 1 and 2 of course, may be used.

To specify the line number N, if N is less than 256, then enter:

```
POKE 17055, N
POKE 17056, 0 (DEFAULT 0)
```

If N is greater than 255 then enter:

```
POKE 17055, N — 256 * INT (N/256)
POKE 17056, INT (N/256)
```

Call the program with Rand USR 17254. When a Rem is used for the first time, key 1 must be pressed next to enter the edit screen mode. Subsequently, when a question mark appears in the bottom left-hand corner one of the following 1 to 6 functions may be selected by pressing the corresponding number key and 7 by pressing Newline.

1. Edit screen
2. Erase and move block
3. Move block without erase
4. Erase block
5. Erase screen
6. Print block
7. Return to Basic

Function 1 enables the editing of the whole screen, definition of the block size and copying the block into a Rem. The cursor position is shown by a flashing character, a blank will flash as a black square. This position may be moved anywhere on the screen using the cursor control keys with shift.

Moving horizontally, on reaching the end of a line, the next position will be at the opposite end of the next line vertically, depending on the direction of travel. Initially, when a character is entered the cursor will move to the next print position thus allowing faster input of text and data and also resulting in character repeat at flash speed if the key is held down.

By pressing shift and function together, this automatic increment is cancelled and movement is by cursor control keys only. This will enable easier construction of graphic figures etc. Pressing shift and function again will return to automatic increment. The cursor will not move out of the confines of the screen.

The mode letter will appear in the bottom left-hand corner of the screen. Initially N, this indicates that the normal single-character face values of all keys are available. Pressing shift and graphics characters will have those as face value.

Shift and rub out will give an inverse I, indicating that all keys are now the inverse of face value. Shift and edit will return mode to N. The mode may be changed at any stage and in any sequence.

The definition of the print position and block size is achieved as follows. To the right of the mode letter is a number, 1-3, which indicates the next print position or block size value that is to be defined. A1 indicates the top left-hand corner of the block. To define this, move the cursor to the required position and press Newline. A2 will now appear, indicating that following this last procedure you may define the top right-hand corner of the block. A2 then indicates that you may define the height or number of lines in the block and only vertical movement is necessary for this.

Once the 3 has appeared, editing the screen can be carried out at any stage up to Newline,



when the block is copied into the designated Rem which is adjusted in size to suit.

Once entered, the whole procedure must be completed. Any attempt to define impossible positions in stages 2 and 3 will result in a return to stage 1, but all data on screen will be retained. The block size may range from a single character through any rectangular form to a full screen. After Newline on 3 there is a return to a choice of function.

On pressing the key, for function 2, "M.E." will appear in the bottom left-hand corner and the top left-hand corner character of the block will flash. Using Shift and the cursor keys the whole block is moved and the previous position is erased.

The block will disappear off the right-hand edge of the screen until the flashing character is at the extreme right. The next move will result in the whole block being printed one line down at the far left of the screen.

With left-hand movement the reverse will

(continued on next page)

Listing 2. 1 Rem.

```
16514 1D011E021F872004 =  264
16522 2105228323032485 =  410
16530 36813C822A073784 =  609
16538 39063E862608380A =  371
16546 29092B8A2C892D88 =  593
16554 2A0C40233E33CDF5 =  716
16562 41CD29411803CD1A =  634
16570 41FE7520F9CD2241 = 1021
16578 3EAECDF541CD2941 = 1062
16586 1803CD1A41FE7720 =  728
16594 03CD2241E63FC680 =  926
16602 18F03EACCDF541CD = 1218
16610 29411803CD1A41FE =  683
16618 742003CD2241E506 =  690
16626 1621814023BE2328 =  548
16634 0410F918017EE118 =  669
16642 E3211C413E00BE20 =  637
16650 023E0C77E11818C1 =  661
16658 18C8C11897C118A8 =  977
16666 771800CDA9411807 =  613
16674 E301070 0ED42E3CD =  970
16682 7941E5CD9B4128DC = 1100
16690 FE7928CDE147C640 = 1178
16698 7838EC112100FE70 =  828
16706 2821FE712822FE72 =  882
16714 2823FE732824FE74 =  890
16722 28BDFE7528BCFE77 = 1201
16730 28BB3DFE7528023C =  761
16738 C9C1C9CDE44118BF = 1308
16746 CDD64118BACDBE41 = 1154
16754 18B5CDA94118B07E =  970
16762 F5FE0028043E0018 =  629
16770 023E80770E300600 =  379
16778 10FE0D20F9F1770E =  938
16786 30060010FE0D20F9 =  618
16794 C9CDBB02444D5114 =  841
16802 C8CDBD077E15C9FD = 1202
16810 7E3EFE1F2009CDD3 =  930
16818 41D823FD363EFF23 =  975
16826 FD343EC9FD7E3EFE = 1263
16834 002009CDE141D82B =  795
16842 FD363E202BFD353E =  812
16850 C9110000FD7E3DC6 =  856
16858 EBD819FD343DC911 = 1060
16866 0000FD7E3DD601D8 =  871
16874 ED52FD353DC901FA = 1138
16882 02180301F802E52A =  551
16890 0C400977E1C93E15 =  713
16898 4690D8233E1F4690 =  772
16906 D8237E060190D83E =  806
16914 204690D8237E0601 =  630
16922 90D83E164690C9CD = 1064
16930 A142E506047E3DFE =  907
16938 75286A2310F7E1E5 = 1015
16946 CD004230052E16CD =  597
16954 BF42E146E5112100 =  831
16962 2A0C402378FE0028 =  567
16970 031910FDEBE1234E =  870
16978 EB09EB47234E237E =  824
16986 23C9CD2142F5D504 = 1002
16994 C506007EFE76282B =  784
17002 EDA0E280421A3DFE = 1158
17010 75280218EE41237E =  647
17018 FE76281710F8C1D1 = 1101
17026 F1EBD511210019EB =  999
17034 E13DC8F53E159038 = 1014
17042 0418CBC1C1C1C9   = 1011
```

(continued from previous page)
occur. Also the block will disappear from the bottom of the screen, but only until the top line is on the bottom line of the screen. The print position is automatically altered so subsequent printing of the block will be in the last position. If the block is only partly shown on screen, the whole block is retained.

Newline will exit from the routine and return to the choice of function.

Function 3 is the same as function 2 except that the previous position is not erased. An M will appear at the bottom left-hand corner. Function 4 erases only the specified block. Function 5 fast-erases the whole screen. Function 6 prints the block at the specified position. Function 7 exits from the routine and returns to Basic control.

Delete the hex loader

At this stage, if all of the routines have been checked then the hex loader program may be deleted but do not remove the line following line 2.

Several of the routines in the program may be used directly and therefore be of assistance in programs. Their addresses are as follows:

16554 Screen edit and block definition only
16988 Prints block
17057 Returns address of the first character in the designated Rem
17197 Clears screen
17216 Erases block

The following are useful addresses:

17254 Start of main program
17055, 17056 Designate Rem — low, high
16775, 16786 Flash speed — off, on

The Rem statements containing a block will have an extra four characters at the beginning. The first two define the top left-hand corner of the block and use the same values as a Print At command. The third is the length of the line, width, and the fourth is the number of lines, height.

These numbers may be altered directly and a routine is included to find the address of the first number, therefore

LET X = USR 17057

will give X that value and so Poke X, Peek (X + 1) will print the block one step down the next time the print routine is called and Poke (X+1), Peek (X+1) + 1 will print the block one step to the right. A blank line included when the block is defined will provide trailing blanks to erase the previous position. Out-of-range values will stop the program and give an error N.

Listing 4. Hex check.

```
290   GOSUB 240
300   FOR A=S TO F STEP 8
310   SCROLL
320   PRINT A; "  ";
330   IF INKEY$ = "" THEN GOTO
      330
340   FOR G=0 TO 7
350   LET P=PEEK (A+G)
360   LET B=INT (P/16)
370   LET C=P-16*B
380   LET A$=CHR$ (B+28) +CHR$ (C+
      28)
390   PRINT A$;
400   NEXT G
410   NEXT A
```



The flash speed may be altered to suit you. The byte address 16775 controls the character off time and 16786 the on time. Initially both have the value 48 but may be Poked with any number between 1 and 255 — the larger, the slower.

Controlled by timing

Other functions are controlled by this timing and so too fast a speed may result in loss of program control. Although the program is essentially crash-proof, beware of the following. When using a Rem for the first time, always use the Edit screen mode, key 1, first and do not use the program while scroll is effective.

Cassette including a copy of Screen and a full renumber program may be obtained for £3 from E M Stenlake, 162 Brooks Lane, Whitwick, Leicester LE6 4DF.

Listing 3. 2 Rem.

```
17055 0000217D40ED5B9F =  709
17063 4246234EEBA7ED42 =  954
17071 2816EB234E234609 =  524
17079 237EFE7620E72E1A =  868
17087 FD7500ED7B0240C9 =  997
17095 210300193EEABE20 =  579
17103 ED23444DC93E2ACD =  927
17111 F041060018020 6C9 =  544
17119 213F43703E32CDF5 =  837
17127 41CD5C42CDA1424E =  938
17135 2346ED433D40CD21 =  772
17143 42EBCD7941E5CD9B = 1281
17151 41E128F6112100FE =  880
17159 70CCE441FE71CCD6 = 1394
17167 41FE72CCBE41FE73 = 1261
17175 CCA9413DFE75C8CD = 1275
17183 3F43CDA142ED4B3D =  935
17191 4071237018BB2A0C =  589
17199 40230E1606203600 =  227
17207 2310FB230DC818F4 =  818
17215 00CD2142EB041121 =  593
17223 00F5C541E5360023 =  825
17231 7EFE76280210F6E1 = 1027
17239 19C1043E16903805 =  511
17247 F13DC818E4C1C9CD = 1353
17255 5C42210000223D40 =  350
17263 3E00CDF0413E0FCD =  854
17271 F541CD7941E5CD9B = 1290
17279 41E128F6FE1DCCA7 = 1230
17287 43FE1ECCD442FE1F = 1118
17295 CCDD42FE20CC4043 = 1112
17303 FE21CC2D43FE2228 =  931
17311 C63DFE7520C4C9E1 = 1284
17319 210000223D40CD5C =  489
17327 423E1DCDF041CDAA = 1042
17335 40E5FD4E3EFD463D = 1070
17343 C53E1ECDF041CDAE = 1178
17351 40C1FD7E3E9138D7 = 1114
17359 5714C5D53E1FCDF0 = 1055
17367 41CDAE40D1C1FD7E = 1289
17375 3D9038C33C5FD5C5 = 1021
17383 435A160021000019 =  237
17391 10FD1E0619E5E522 =  822
17399 3F40CDA1422B2B46 =  715
17407 2B4EED434140E3ED = 1018
17415 4B1C400923A7ED72 =  729
17423 300AE1D1732372CD =  961
17431 230F1807060A3310 =  164
17439 FD1884ED4B41402A =  892
17447 3F40A7ED42E5282A =  908
17455 E5F5CDA1422A4140 = 1077
17463 092B2BEB2A1C40A7 =  631
17471 ED52444D03F1380A =  774
17479 19D1EB19EB03EDB8 = 1153
17487 1808E1D5EB19D1EB = 1174
17495 EDB0010209210C40 =  534
17503 5E235623E5C5D5CD = 1094
17511 A142D1C1A7ED5230 = 1163
17519 0FE1E3E519EBE1E3 = 1408
17527 2B722B7323231801 =  410
17535 E110DD2129400D06 =  619
17543 0120D5CDA142C1D1 = 1080
17551 722373D123722373 =  772
17559 C506004A7B23D1E3 =  871
17567 19D1C5E5EDB00121 = 1107
17575 00E109C13D20F3CD =  968
17583 2B0FC9           =  259
```

QUICKPLOT

Program 1.

```
10 REM  *************************
20 REM  **     C64 QUICKPLOT    **
30 REM  **    G.HATTO  JAN'85   **
40 REM  *************************
50 :
60 :
70 POKE631,131:POKE198,1
80 POKE64*256,0:POKE44,64:NEW
```

Program 2.

```
10 REM  *************************
20 REM  **   QUICKPLOT LOADER   **
30 REM  **   G.HATTO   JAN'85   **
40 REM  *************************
50 :
60 :
70 MAXNUMBER = 48
80 DEF FN LINENUMBER(A) = PEEK(63)+PEEK(
   64)*256
90 :
100 READ ADDRESS
110 FOR COUNT=1 TO MAXNUMBER:GOSUB 230:R
    EAD CHECKSUM
120 IF SUM <> CHECKSUM THEN PRINT"CHECKS
    UM ERROR IN ";FN LINENUMBER(0):END
130 L = FN LINENUMBER(0):PRINT L
140 IF ADDRESS <> L+8 THEN PRINT"LINE NU
    MBER";L;" NOT IN ORDER":END
150 NEXT COUNT
160 :
170 PRINT:PRINT"   HIRES ="
180 PRINT"       SYS4096,INK,PAPER,GCLEAR
    "
190 PRINT:PRINT"   PLOT  ="
200 PRINT"       SYS4318,X,Y,PLOT/UNPLOT"
210 END
220 :
230 SUM = 0
240 FOR I=1 TO 8:READ ENTRY$
250 GOSUB 310
260 POKE ADDRESS,ENTRY
270 ADDRESS = ADDRESS+1:SUM = SUM+ENTRY
280 NEXT I
290 RETURN
300 :
310 IF LEN(ENTRY$) <> 2 THEN 400
320 A$ = RIGHT$(ENTRY$,1):GOSUB 360:ENTR
    Y = V
330 A$ = LEFT$ (ENTRY$,1):GOSUB 360:ENTR
    Y = ENTRY+V*16
340 RETURN
350 :
360 IF A$<"0" OR A$>"F" OR (A$>"9" AND A
    $<"A") THEN 400
370 V = ASC(A$)-48+7*(A$>"9")
380 RETURN
390 :
400 PRINT"DATA ERROR IN ";FN LINENUMBER(
    0):END
410 :
420 DATA 4096
4096 DATA AD,86,02,85,9B,AD,21,D0,1011
4104 DATA 29,0F,85,9C,20,19,10,20,450
4112 DATA 62,10,20,9D,10,20,8A,10,505
4120 DATA 60,A0,00,A9,2C,D1,7A,D0,1008
4128 DATA 24,20,9B,B7,86,9B,A0,00,855
4136 DATA A9,2C,D1,7A,D0,17,20,9B,962
4144 DATA B7,8A,29,0F,05,9C,A0,00,826
4152 DATA A9,2C,D1,7A,D0,07,20,9B,946
4160 DATA B7,E0,00,F0,03,20,49,10,771
4168 DATA 60,A9,00,A0,20,85,4E,84,800
4176 DATA 4F,A9,00,AA,A8,91,4E,C8,1009
4184 DATA D0,FB,E6,4F,E8,E0,20,D0,1464
4192 DATA F4,60,A5,9B,0A,0A,0A,0A,700
4200 DATA 05,9C,A0,00,99,00,08,99,635
4208 DATA 00,09,99,00,0A,99,00,0B,336
4216 DATA C8,D0,F1,60,A9,14,8D,18,1099
4224 DATA D0,AD,11,D0,29,DF,8D,11,1028
4232 DATA D0,60,AD,18,D0,29,08,09,767
4240 DATA 28,8D,18,D0,AD,11,D0,09,820
4248 DATA 20,8D,11,D0,60,78,A9,D5,996
4256 DATA A0,10,8D,00,03,8C,01,03,464
4264 DATA A9,B4,A0,10,8D,14,03,8C,829
4272 DATA 15,03,58,60,20,8A,10,4C,518
4280 DATA 31,EA,A5,C5,C5,FE,85,FE,1483
4288 DATA F0,0E,C9,03,D0,0A,AD,11,866
4296 DATA D0,29,20,F0,04,20,7C,10,697
4304 DATA 60,20,8A,10,60,8A,38,03,567
4312 DATA 20,7C,10,4C,8B,E3,20,FD,899
4320 DATA AE,20,EB,B7,86,9B,20,F1,1186
4328 DATA B7,86,02,38,A9,C7,E5,9B,1127
4336 DATA D0,05,A2,0E,4C,D5,10,85,795
4344 DATA 9C,4A,4A,4A,A8,A5,15,4A,806
4352 DATA D0,F0,A5,14,90,06,2C,C2,1021
4360 DATA E1,D0,E7,18,29,F8,79,38,1154
4368 DATA 11,85,4E,A5,15,79,51,11,633
4376 DATA 85,4F,A5,9C,29,07,A8,A5,914
4384 DATA 14,29,07,AA,A5,02,F0,08,653
4392 DATA BD,6A,11,11,4E,91,4E,60,726
4400 DATA BD,72,11,31,4E,91,4E,60,766
4408 DATA 80,40,80,C0,00,40,80,C0,768
4416 DATA 00,40,80,C0,00,40,80,C0,768
4424 DATA 00,40,80,C0,00,40,80,C0,768
4432 DATA 00,20,21,22,23,25,26,27,240
4440 DATA 28,2A,2B,2C,2D,2F,30,31,358
4448 DATA 32,34,35,36,37,39,3A,3B,438
4456 DATA 3C,3E,80,40,20,10,08,04,374
4464 DATA 02,01,7F,BF,DF,EF,F7,FB,1281
4472 DATA FD,FE,00,00,00,00,00,00,507
```



WHEN A BASIC error occurs in the high-resolution mode on some Basic extension packages, the resulting error message is displayed in an unreadable form because of the way in which the video chip works. This can be a real problem when developing Basic programs. It is also difficult to switch between one display mode and another withut remembering a long series of Pokes.

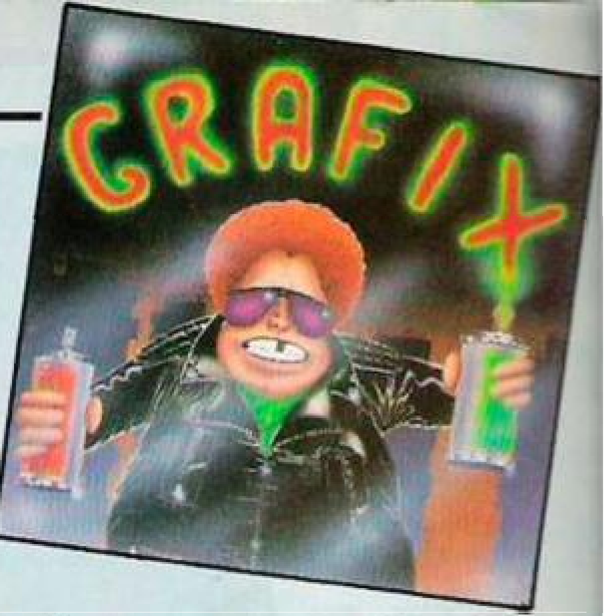
The following machine-code program runs

Assembly listing.

```
10 ;PROG       C64 QUICKPLOT   G.HATTO
20 ;           ---------------------------
30 ;
40 ;                          20/1/85
50 ;
60 ;
70 ;
80 ;           PROVIDES A PLOT COMMAND
90 ;           IN  BASIC TOGETHER WITH
100 ;          FULL  CONTROL  OVER THE
110 ;          HI-RESOLUTION SCREEN.
120 ;
130 ;
140            ORG $1000
150 ;
160 ;
170 BASEADDR   EQU $2000
180 HSCREEN    EQU $0800
190 VIDEOCHIP  EQU $D000
200 ;
210 ;
220 ;          **** HIRES ****
230 ;
240 ;
250 ;          SETS UP HIGH-RESOLUTION
260 ;          SCREEN AND  ENABLES THE
270 ;          MODE KEY (F7) AND ERROR
280 ;          HANDLER.
290 ;
300 ;
310 COL        EQU $0286
320 INK        EQU $9B
330 PAPER      EQU $9C
340 ;
350 ;
360 HIRES      LDA COL
370            STA INK
380            LDA VIDEOCHIP+$21
390            AND #$0F
400            STA PAPER
410 ;
420            JSR GETPARAMS
430            JSR SETCOLRS
440            JSR SETVECTS
450            JSR SETBITMAP
460            RTS
470 ;
480 ;
490 ;
500 ;          **** GETPARAMS ****
510 ;
520 ;
530 ;          GETS PARAMETERS FOR THE
540 ;          INK,  PAPER  AND  HIGH-
550 ;          RESOLUTION SCREEN CLEAR
560 ;
570 ;
580 TXTPTR     EQU $7A
590 GETBYTE    EQU $B79B
600 ;
610 ;
620 GETPARAMS  LDY #$00
630            LDA #',
640            CMP (TXTPTR),Y
650            BNE GCLR
660 ;
670            JSR GETBYTE
680            STX INK
690 ;
700            LDY #$00
710            LDA #',
720            CMP (TXTPTR),Y
730            BNE GCLR
740 ;
750            JSR GETBYTE
760            TXA
770            AND #$0F
780            STA PAPER
790 ;
800            LDY #$00
810            LDA #',
820            CMP (TXTPTR),Y
830            BNE GCLR
840 ;
850            JSR GETBYTE
860            CPX #$00
870            BEQ GPEXIT
880 ;
890 GCLR       JSR GCLEAR
900 GPEXIT     RTS
910 ;
920 ;
930 ;
940 ;          **** GCLEAR ****
950 ;
960 ;
970 ;          CLEARS  HIGH-RESOLUTION
980 ;          SCREEN.
990 ;
1000 ;
1010 POINTER   EQU $4E
1020 ;
1030 ;
1040 GCLEAR    LDA #BASEADDR
1050           LDY ↑BASEADDR
1060           STA POINTER
1070           STY POINTER+$01
1080 ;
1090           LDA #$00
1100           TAX
1110           TAY
1120 LCLR      STA (POINTER),Y
1130           INY
1140           BNE LCLR
1150           INC POINTER+$01
1160           INX
1170           CPX #$20
1180           BNE LCLR
1190           RTS
1200 ;
1210 ;
1220 ;
1230 ;         **** SETCOLRS ****
1240 ;
1250 ;
1260 ;         SETS   COLOURS  ON  THE
1270 ;         HIGH-RESOLUTION SCREEN.
1280 ;
1290 ;
1300 SETCOLRS  LDA INK
1310           ASL
1320           ASL
1330           ASL
1340           ASL
1350           ORA PAPER
1360 ;
1370           LDY #$00
1380 NXTCOLR   STA HSCREEN,Y
1390           STA HSCREEN+$100,Y
1400           STA HSCREEN+$200,Y
1410           STA HSCREEN+$300,Y
1420           INY
1430           BNE NXTCOLR
1440           RTS
1450 ;
1460 ;
1470 ;
1480 ;         **** SCREENDEF ****
1490 ;
1500 ;
1510 ;         SETS DEFAULT TEXT MODE.
1520 ;
1530 ;
1540 SCREENDEF LDA #$14
1550           STA VIDEOCHIP+$18
1560           LDA VIDEOCHIP+$11
1570           AND #$DF
1580           STA VIDEOCHIP+$11
1590           RTS
1600 ;
1610 ;
1620 ;
1630 ;         **** SETBITMAP ****
1640 ;
1650 ;
1660 ;         SETS BITMAP MODE.
1670 ;
1680 ;
1690 SETBITMAP LDA VIDEOCHIP+$18
1700           AND #$08
1710           ORA #$20
1720           STA VIDEOCHIP+$18
1730 ;
1740           LDA VIDEOCHIP+$11
1750           ORA #$20
1760           STA VIDEOCHIP+$11
1770           RTS
1780 ;
1790 ;
1800 ;
1810 ;         **** SETVECTS ****
1820 ;
1830 ;
1840 CINV      EQU $0314
1850 IERR      EQU $0300
1860 ;
1870 ;
1880 SETVECTS  SEI
1890           LDA #ERR
1900           LDY ↑ERR
1910           STA IERR
```

Program 3.

```
10 REM  ****************************
20 REM  *                          *
30 REM  *   QUICKPLOT DEMO  GH'85  *
40 REM  *                          *
50 REM  *  ITERATIVE SOLUTION OF   *
60 REM  *      1-2*Y*Y/(X*X)       *
70 REM  *                          *
80 REM  ****************************
100 :
110 HIRES=4096:PLOT=4318
120 POKE53280,6:SYS HIRES,1,6,1
130 DEF FNA(X)=1-2*Y*Y/(X*X)
140 :
150 FOR X=0 TO 319 STEP 0.1
160 Y=FNA(X/1000+1)
170 SYS PLOT,X,100+85*Y,1
180 NEXT X
190 END
```





Geoff Hatto helps you plot points and switch quickly between hires and character displays.

on a standard Commodore 64 computer and allows Basic to plot pixels on a high-resolution screen without any of these problems described. The first routine sets up the high-resolution display, IRQ and error handler. The second is the plot/unplot routine itself. To increase the overall speed of the pixel plotting, I chose a direct call to the machine-code routines, rather than trying to implement two new commands in Basic.

To initialise the high-resolution screen and the mode key — function key F7 — the following command is required;

SYS 4096, ink, paper, gclear

Where ink and paper are the colour values 0 to 15. A non-zero value for gclear will clear the high-resolution screen, a value of zero will leave the current screen intact. These parameters may be completely or partly eliminated to obtain default values — that is, SYS4096 would assume current screen editor colours and would cl.ear the existing high-resolution screen. Gclear is assumed to be 1.

After this command has beren issued, the function key F7 allows switching in and out of high-resolution mode — even if a Basic program is running. If any Basic error should occur, the screen will automatically revert to text mode, displaying the error message.

The plot command has tghe usual syntax, although all of the parameters must be specified:

SYS4318,x,y,p

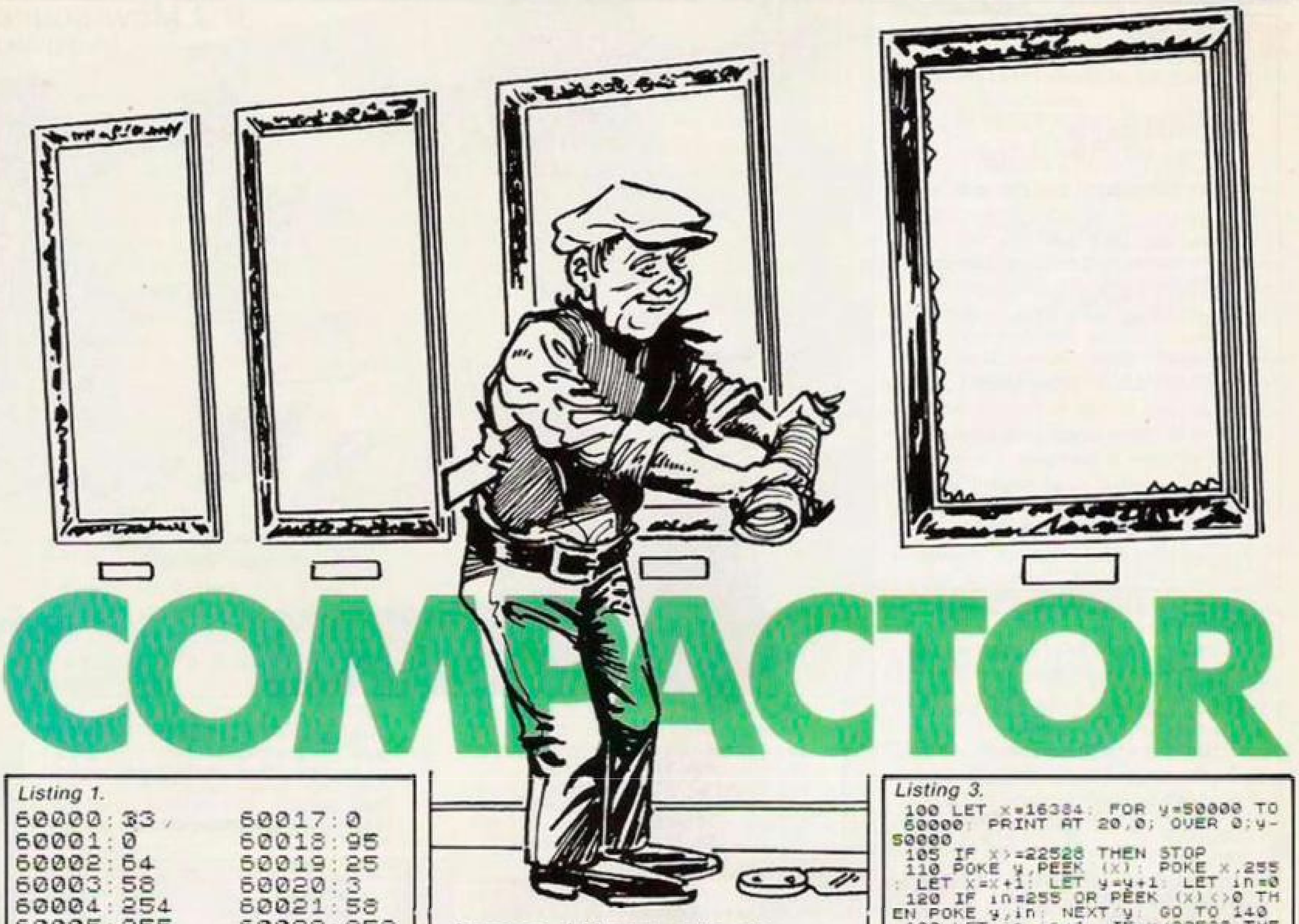
Where x and y are the pixel co-ordinates and p is either zero, (unplot), or non-zero, plot.

To enter Quickplot, type in program 1 and save it, followed by program 2. Program 1 automatically loads and runs program 2. Simple error-checking is performed on program 2's data statements. If typing errors are found, they are reported and must be corrected. The final copy of program 2 is then resaved on tape.

Program 3 demonstrates the use of both the Hires and Plot facilities given by the quick-plot program. The demonstration will only work if programs 1 and 2 have been previously loaded and executed.

An assembly listing of the main program is also included and has been written in a structured manner to allow for future program development. ■

```
1920            STY IERR+1
1930 ;
1940            LDA #IRQ
1950            LDY >IRQ
1960            STA CINV
1970            STY CINV+1
1980            CLI
1990            RTS
2000 ;
2010 ;
2020 ;
2030 ;          **** IRQ ****
2040 ;
2050 ;
2060 ;          MAIN IRQ ROUTINE.
2070 ;
2080 ;
2090 STDIRQ     EQU $EA31
2100 ;
2110 ;
2120 IRQ        JSR F7CHECK
2130 IRQEXIT    JMP STDIRQ
2140 ;
2150 ;
2160 ;
2170 ;          **** F7CHECK ****
2180 ;
2190 ;
2200 ;          CHECKS FOR MODE KEY AND
2210 ;          TOGGLES BETWEEN  BITMAP
2220 ;          AND TEXT MODE IF KEYED.
2230 ;
2240 ;
2250 LSTX       EQU $C5
2260 LSTKEY     EQU $FE
2270 ;
2280 ;
2290 F7CHECK    LDA LSTX
2300            CMP LSTKEY
2310            STA LSTKEY
2320            BEQ NOFKEY
2330            CMP #$03
2340            BNE NOFKEY
2350            LDA VIDEOCHIP+$11
2360            AND #$20
2370            BEQ SETBMM
2380 ;
2390            JSR SCREENDEF
2400 NOFKEY     RTS
2410 ;
2420 SETBMM     JSR SETBITMAP
2430            RTS
2440 ;
2450 ;
2460 ;
2470 ;          **** ERR ****
2480 ;
2490 ;
2500 ;          MAIN ERROR HANDLER.
2510 ;
2520 ;
2530 ERR        TXA
2540            BMI NOERR
2550            JSR SCREENDEF
2560 ;
2570 NOERR      JMP $E38B
2580 ;
2590 ;
2600 ;
2610 ;          **** PLOT ****
2620 ;
2630 ;
2640 ;          PLOTS A PIXEL ON THE
2650 ;          HIGH-RESOLUTION SCREEN.
2660 ;
2670 ;
2680 BS         EQU BASEADDR DIV $100
2690 PLOTST     EQU $02
2700 XL         EQU $14
2710 XH         EQU $15
2720 Y          EQU $9B
2730 Y1         EQU $9C
2740 MEM        EQU $4E
2750 CHECKCOM   EQU $AEFD
2760 GETPLOTST  EQU $B7F1
2770 GETXY      EQU $B7EB
2780 MASK       EQU $E1C2
2790 ;
2800 ;
2810 ;          BASIC ENTRY POINT.
2820 ;
2830 ;
2840 BASICPLOT  JSR CHECKCOM
2850            JSR GETXY
2860            STX Y
2870            JSR GETPLOTST
2880            STX PLOTST
2890 ;
2900 ;
2910 ;          MACHINE CODE ENTRY.
2920 ;
2930 ;
2940 PLOT       SEC
2950            LDA #$C7
2960            SBC Y
2970            BCS PTTBLE
2980 ;
2990 ILLQUANT   LDX #$0E
3000            JMP ERR
3010 ;
3020 PTTBLE     STA Y1
3030            LSR
3040            LSR
3050            LSR
3060            TAY
3070 ;
3080            LDA XH
3090            LSR
3100            BNE ILLQUANT
3110 ;
3120            LDA XL
3130            BCC NOTH
3140            BIT MASK
3150            BNE ILLQUANT
3160            CLC
3170 ;
3180 NOTH       AND #$F8
3190            ADC YTABLEL,Y
3200            STA MEM
3210            LDA XH
3220            ADC YTABLEH,Y
3230            STA MEM+1
3240 ;
3250            LDA Y1
3260            AND #$07
3270            TAY
3280 ;
3290            LDA XL
3300            AND #$07
3310            TAX
3320 ;
3330            LDA PLOTST
3340            BEQ UNPLT
3350
3360            LDA TBIT1,X
3370            ORA (MEM),Y
3380            STA (MEM),Y
3390            RTS
3400 ;
3410 UNPLT      LDA TBIT2,X
3420            AND (MEM),Y
3430            STA (MEM),Y
3440            RTS
3450 ;
3460 ;
3470 YTABLEL    DFB $00,$40,$80,$C0
3480            DFB $00,$40,$80,$C0
3490            DFB $00,$40,$80,$C0
3500            DFB $00,$40,$80,$C0
3510            DFB $00,$40,$80,$C0
3520            DFB $00,$40,$80,$C0
3530            DFB $00
3540 ;
3550 ;
3560 YTABLEH    DFB BS,BS+1
3570            DFB BS+2,BS+3
3580            DFB BS+5,BS+6
3590            DFB BS+7,BS+8
3600            DFB BS+10,BS+11
3610            DFB BS+12,BS+13
3620            DFB BS+15,BS+16
3630            DFB BS+17,BS+18
3640            DFB BS+20,BS+21
3650            DFB BS+22,BS+23
3660            DFB BS+25,BS+26
3670            DFB BS+27,BS+28
3680            DFB BS+30
3690 ;
3700 ;
3710 TBIT1      DFB $80,$40,$20,$10
3720            DFB $08,$04,$02,$01
3730 ;
3740 ;
3750 TBIT2      DFB $7F,$BF,$DF,$EF
3760            DFB $F7,$FB,$FD,$FE
3770 ;
3780 ;
3790 ;
```



COMPACTOR

James Higgo steals space on the Spectrum.

Listing 1.

```
60000:33     60017:0
60001:0      60018:95
60002:64     60019:25
60003:58     60020:3
60004:254    60021:58
60005:255    60022:252
60006:71     60023:255
60007:56     60024:184
60008:255    60025:32
60009:255    60026:240
60010:79     60027:56
60011:10     60028:253
60012:119    60029:255
60013:3      60030:185
60014:10     60031:32
60015:60     60032:234
60016:22     60033:201
```

Listing 2.

```
10 LET t=0
20 FOR x=60000 TO 60033
30 INPUT "value ";(x);":";a
40 POKE x,a
50 PRINT x;":";a
60 LET t=t+a
70 NEXT x
80 IF t<>3968 THEN PRINT "ERROR
- start again"
```

Listing 3.

```
 100 LET x=16384: FOR y=50000 TO
 60000: PRINT AT 20,0; OVER 0;y-
50000
 105 IF x>=22528 THEN STOP
 110 POKE y,PEEK (x): POKE x,255
: LET x=x+1: LET y=y+1: LET in=0
 120 IF in=255 OR PEEK (x)<>0 TH
EN POKE y,in: NEXT y: GO TO 140
 130 LET in=in+1: IF x<22528 THE
N LET x=x+1: GO TO 120
 140 PRINT "ended"
 145 POKE y+1,66: POKE y+2,66: P
OKE y+3,66
 150 STOP
 200 REM
              LINES 200 ONWARD PERF
ORM THE SAME FUNCTION AS THE MAC
HINE CODE ROUTINE BUT MUCH SLOWE
R AND BASIC
 205 CLS : LET xs=16384: FOR x=5
0000 TO 60000
 210 POKE xs,PEEK (x)
 220 LET xs=xs+PEEK (x+1)+1
 225 IF xs>=22528 THEN STOP
 230 LET x=x+1: NEXT x
```

A FEW MONTHS ago, several routines which stored a screen in memory and then recalled it were published. This takes up much memory, and only three screens may be stored on a 48K Spectrum in this way. This routine compacts the screen by not storing all the blank spaces, but by counting up how many of them there are. The resulting data is stored by a Basic program, and then recalled from any area of memory by the machine-code routine.

The advantages of this technique are that it takes up about a tenth of the memory for a normal picture, is faster to draw, and several images may be superimposed.

To store a picture, the Basic program must be run — use Goto 100, as Run will wipe the screen. The compacted data will be put into locations 50000 onwards, depending on the complexity of the picture.

The length of the data will be indicated by the final value of the number at the bottom left-hand corner of the screen. The data may then be saved using

SAVE "data" CODE 50000

and the length number.

Once the data has been saved it may be reloaded into a new location using

LOAD "data" CODE

and the value of the desired location.

To recall the picture, the machine-code routine must be loaded, and the values of the start and finish addresses of the data must be Poked into the locations 65532 to 65535, by converting the number to hex and Poking the high and low bytes as follows:

65532: end location high byte
65533: end location low byte
65534: start location high byte
65533: start location low byte

For example, if the data is left at location 50000 — not saved and reloaded —

POKE 65532,200 end locations (ave. picture length)
POKE 65533,180
POKE 65534,195
POKE 65535,000 start locations

Using these values for the end locations may chop off or add on something to your picture if it is longer or shorter than the average. Once this had been done, to recall the picture, you need only type

RANDOMISE USR 60000

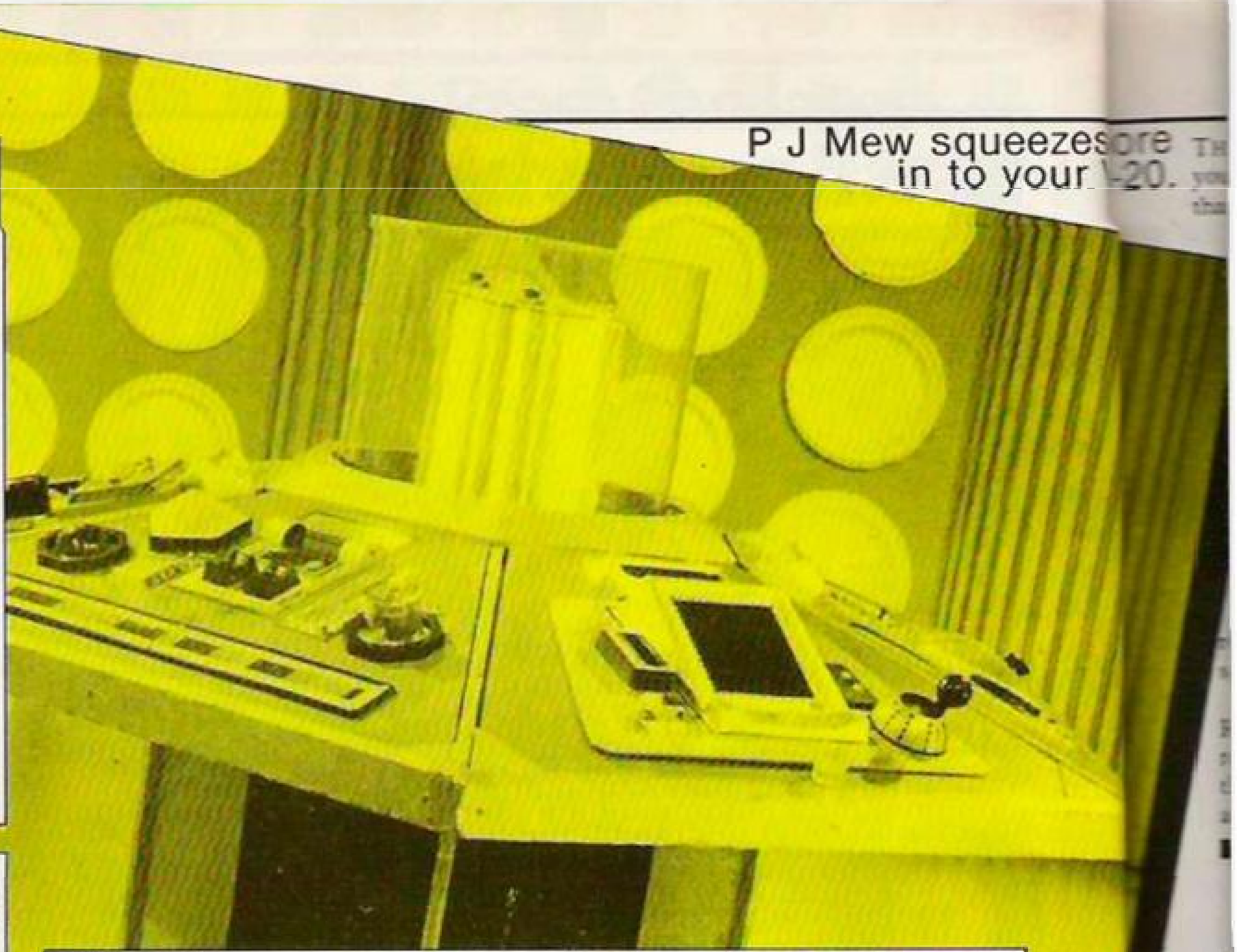
Listing 1 is the data for the machine-code routine. Listing 2 is a loader for listing 1. Listing 3 is the compacting program.

Here's how to enter the program:

1. Type in machine-code loader and enter data by running it
2. Delete the loader line by line, and type in the compacter program
3. Draw a picture, then run the computer by typing Goto 100. Commands for drawing the picture may be entered in lines 1-99, or a picture may be loaded from tape
4. Save the compacted code on tape by typing SAVE "data" CODE 50000 and the length of the data, which is given at the bottom left of the screen after compacting
5. Reload the data into the desired area, and Poke the start and finish locations into stores 65532 to 65534, as described earlier.
6. Type RANDOMISE USR 60000

If the start and end locations have not been calculated correctly, the screen will be filled with a portion of the picture, or garbage. The locations for finish are different for each screen, and are found by hexing the length of the data plus 50000, or whatever the start location is. ■

P J Mew squeezes ore
in to your -20.



Demo listing.

```
10 REM**THIS LINE WILL BE REMOVE
D COMPLETELY*********************
**
20 T=TI
30 IFX>100THEN130
40 X=X+1:GOSUB1000:GOTO30
130 PRINT"■"
140 PRINT"■■■■THIS LINE AND ALL
THE"
150 PRINT"FOLLOWING PRINT
STATEMENTS WILL BE"
160 PRINT"REDUCED TO A SINGLE"
170 PRINT"LINE"
200 T1=TI:FR=FRE(0)
210 PRINT"■■ THIS DEMONSTRATION"

220 PRINT" PROGRAM HAS TAKEN"
230 PRINT(T1-T)/60 "SECONDS TO R
UN"
250 PRINT
260 PRINT"AND USES"19967-FR" BYT
ES"
270 END
1000 REM**THIS LINE   WILL BE RE
DUCED TO THEWORD REM
1010 PRINT"■■■■■■■■■■■■■■■"X:REM*
*THIS REM   WILL GO
1020 RETURN
```

Program 1.

```
1 REM*****************************REM
OVE SPACES P.J.MEW 9/1/85********
****************
5 PRINT"■■       LOADING DATA"
10 I=0:A=0:CS=0
20 READA:IFA>255THEN50
30 POKE24390+I,A:I=I+1:CS=CS+A
40 GOTO20
50 IFI<>183THENPRINT"■■■■    WRO
NG NUMBER OF     ■   DATA STATEME
NTS■":END
60 IFCS<>20583THENPRINT"■■■■
 DATA ERROR■":END
70 PRINT"■■■■       SYS24390
    ■  TO REMOVE SPACES■":END
100 DATA165,122,72,165,123,72,16
5,43
110 DATA133,122,165,44,133,123,3
2,197
120 DATA95,32,197,95,32,197,95,1
65,45
130 DATA197,122,165,46,229,123,1
44,64
140 DATA32,197,95,162,32,32,11,2
01,32
150 DATA251,200,32,203,95,208,6,
32,197
160 DATA95,76,84,95,165,122,133,
36,165
170 DATA123,133,37,32,197,95,201
,32
180 DATA240,249,165,122,133,34,1
65,123
190 DATA133,35,56,165,36,233,1,1
33,122
200 DATA165,37,233,0,133,123,32,
208,95
210 DATA76,93,95,32,51,197,104,1
33,123
220 DATA104,133,122,165,55,164,5
6,133
230 DATA51,132,52,165,45,164,46,
133,47
240 DATA132,48,133,49,132,50,96,
230
250 DATA122,208,2,230,123,160,0,
177
260 DATA122,96,160,0,165,45,197,
34,165
270 DATA46,229,35,144,19,177,34,
145,36
280 DATA230,34,208,2,230,35,230,
36,208
290 DATA2,230,37,76,210,95,56,16
5,36
300 DATA233,1,133,45,165,37,233,
0,133
310 DATA46,96,20583:REMCHECKSUM
```

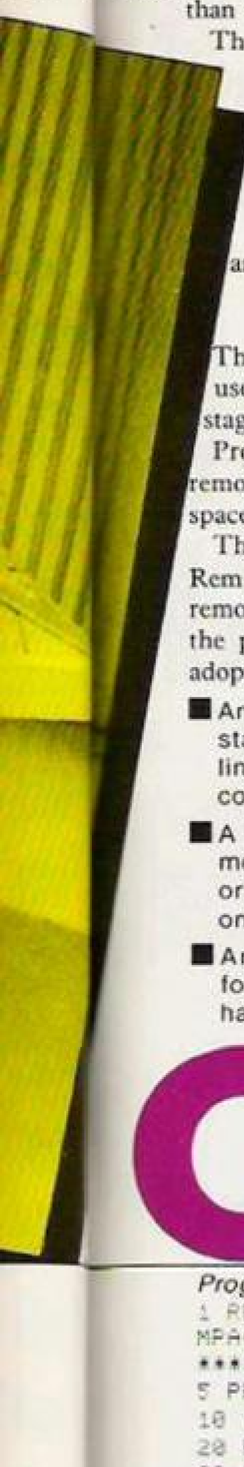
Demo listing.

```
20 T=TI
30 IFX>100THEN130
40 X=X+1:GOSUB1000:GOTO30
130 PRINT"■":PRINT"■■■■THIS LINE
 AND ALL THE":PRINT"FOLLOWING PR
INT       STATEMENTS WILL BE":PR
INT"REDUCED TO A SINGLE":PRINT"L
INE":T1=TI:FR=FRE(0):PRINT"■■ TH
IS DEMONSTRATION":PRINT" PROGRAM
 HAS TAKEN":PRINT(T1-T)/60"SECON
DS TO RUN":PRINT:PRINT"AND USES"
19967-FR" BYTES":END
1000 REM
1010 PRINT"■■■■■■■■■■■■■■■"X:RETU
RN
```

Program 2.

```
1 REM*****************************RE
MOVE REMS P.J.MEW 9/1/85*********
****************
5 PRINT"■■       LOADING DATA"
10 I=0:A=0:CS=0
20 READA:IFA>255THEN50
30 POKE24050+I,A:I=I+1:CS=CS+A
40 GOTO20
50 IFI<>337THENPRINT"■■■■    WRO
NG NUMBER OF     ■   DATA STATEME
NTS■":END
60 IFCS<>39933THENPRINT"■■■■
 DATA ERROR■":END
70 PRINT"■■■■         SYS24050
    ■     TO REMOVE REMS■":END
100 DATA165,122,72,165,123,72,16
5,43
110 DATA133,122,165,44,133,123,3
2,236
120 DATA94,32,236,94,133,3,32,23
6,94
130 DATA133,4,165,45,197,122,165
,46
140 DATA229,123,144,75,32,236,94
,240
150 DATA20,201,143,208,41,32,128
,94
160 DATA165,251,240,15,32,236,94
,32
170 DATA247,94,32,22,95,32,236,9
4,76,0
180 DATA94,56,165,122,233,5,133,
36,165
190 DATA123,233,0,133,37,32,255,
94,76
200 DATA45,94,32,236,94,240,226,
201
210 DATA143,208,247,56,165,122,2
33,1
220 DATA133,122,165,123,233,0,13
3,123
230 DATA76,42,94,32,51,197,104,1
33,123
240 DATA104,133,122,165,55,164,5
6,133
250 DATA51,132,52,165,45,164,46,
133,47
260 DATA132,48,133,49,132,50,96,
165
270 DATA122,72,165,123,72,169,0,
133
280 DATA251,165,43,133,122,165,4
4,133
290 DATA123,32,236,94,32,236,94,
32,236
300 DATA94,165,45,197,122,165,46
,229
310 DATA123,144,47,32,236,94,201
,137
320 DATA240,18,201,141,240,14,20
1,167
330 DATA240,10,201,0,208,227,32,
236,94
340 DATA76,146,94,32,115,0,32,10
7,201
350 DATA165,3,197,20,208,17,165,
4,197
360 DATA21,208,11,169,255,133,25
1,104
370 DATA133,123,104,133,122,96,3
2,242
380 DATA94,201,44,240,220,201,58
,240
390 DATA161,32,236,94,76,146,94,
230
400 DATA122,208,2,230,123,160,0,
177
410 DATA122,96,165,122,133,36,16
5,123
420 DATA133,37,32,9,201,32,251,2
00,165
430 DATA122,133,34,165,123,133,3
5,165
440 DATA36,133,122,165,37,133,12
3,96
450 DATA160,0,165,45,197,34,165,
46,229
460 DATA35,144,19,177,34,145,36,
230,34
470 DATA208,2,230,35,230,36,208,
2,230
480 DATA37,76,24,95,56,165,36,23
3,1
490 DATA133,45,165,37,233,0,133,
46,96,39933:REMCHECKSUM
```

ezes ore our V-20.



THESE THREE programs allow you to compact your Basic programs into much less memory than they would normally take.

The advantages of compressing your programs are that they use up fewer bytes of memory, on average about 10percent less, they run faster because the interpreter does not have to search through so many line numbers, and, because lines longer than 88 characters are almost impossible to edit, a compacted program offers a measure of security against unauthorised tampering.

The three programs printed here may be used individually or together for various stages of text compression.

Program 1 is a simple space remover. It removes all spaces outside quotes but leaves spaces inside quotes intact.

The second program is used for removing Rem statements. However, rather than just remove all Rems regardless of their position in the program, a selective approach has been adopted, and this works as follows:

- Any line containing only a Rem statement and which is not the target line of a Goto or Gosub is removed completely.
- A line containing only a Rem statement but which is the target for a Goto or Gosub, is reduced to the word Rem only, the line number being retained.
- Any line containing Basic text followed by a Rem on the same line, has the Rem removed.

The third and most complex program actually joins the lines of Basic text together to make new lines which can contain up to 250 Basic characters. These 250 characters plus the 2 byte link address, the 2 byte line number and the zero byte at the end of the line make up a total maximum length of 255 bytes per Basic line.

This limit of 255 bytes is imposed by the Rom routine Rechain Lines at $C533 which is used by the operating system during editing and loading and by these programs to relink the new longer lines together.

Lines of Basic cannot just be joined together without taking into account certain conditions which would alter the structure of the program, so the following situations are allowed for:

- Line numbers which are the targets for Gotos and Gosubs are retained.
- Lines containing the keywords If, Then, Goto, Return, End, Rem have nothing added to the ends of them.
- Lines which would become longer than 255 characters are not allowed.

There is no built in protection for the machine code so if you want the code to stay in memory and not be overwritten by Basic it is necessary to enter the following:

POKE56,92:POKE52,92:POKE55,75:
POKE51,75

This lowers the top of Basic below the code.

When the programs have been entered and run and the machine code is in memory, you can if you wish, save the code as a file, saving you the bother of reloading the Basic loaders each time. This is done by entering the following instructions:

POKE43,75:POKE44,92:POKE45,0:
POKE46,96
SAVE'FILENAME',1,1

This saves the area from 23627 to 24576 as a program with a header specifying a forced load, so that any Load instruction will reload the code back into the area it was saved from. After saving in this way it is necessary to enter SYS64802 to reset all the Basic pointers, however the code will stay in memory.

The before and after demonstration programs show some of the features of the compactor programs and if entered and run will indicate a time saving of about 5.5 percent and a reduction in memory useage of about 30 percent.

These programs are designed to locate into the top of Ram on a 16K expanded Vic-20 but can easily be moved elsewhere in memory, for example to start at $A000 if you have a 32K expanded Vic-20, using my program machine-code Mover in the January 1985 issue of *Your Computer*. The programs use a number of the Vic Rom routines between $C000 and $DFFF. If these are isolated and changed to the corresponding Rom locations for the CBM-64, between $S000 and $BFFF the programs should also run on that machine.

In spite of being in machine code the third program in particular may take several minutes to run. ■

COMPACTOR

Program 3.

```
1 REM******************************COMPACTOR P.J.MEW 9/1/85****************************
5 PRINT"         LOADING DATA"
10 I=0:A=0:CS=0
20 READA:IFA>255THEN50
30 POKE23630+I,A:I=I+1:CS=CS+A
40 GOTO20
50 IFI<>417THENPRINT"         WRONG NUMBER OF      DATA STATEMENTS":END
60 IFCS<>49685THENPRINT"      DATA ERROR":END
70 PRINT"          SYS23630       TO COMPACT BASIC":END
100 DATA165,122,72,165,123,72,165,43
110 DATA133,122,165,44,133,123,169,0
120 DATA133,252,32,150,93,32,150,93,32
130 DATA150,93,32,150,93,32,9,201,200
140 DATA132,252,165,45,197,122,165,46
150 DATA229,123,176,3,76,250,92,162,0
160 DATA134,253,189,233,93,170,32,11
170 DATA201,177,122,208,96,166,253,232
180 DATA224,6,208,236,32,9,201,32,251
190 DATA200,165,122,133,36,165,123,133
200 DATA37,32,150,93,133,3,32,150,93,5
210 DATA3,240,75,32,150,93,133,3,32
220 DATA150,93,133,4,32,206,93,165,252
230 DATA240,172,32,24,93,165,251,240,7
240 DATA169,0,133,252,76,108,92,165
250 DATA122,133,34,165,123,133,35,169
260 DATA58,145,36,230,36,208,2,230,37
270 DATA165,36,133,122,165,37,133,123
280 DATA32,161,93,76,114,92,32,9,201
290 DATA32,251,200,32,150,93,76,96,92
300 DATA32,51,197,104,133,123,104,133
310 DATA122,165,55,164,56,133,51,132
320 DATA52,165,45,164,46,133,47,132,48
330 DATA133,49,132,50,96,165,122,72
340 DATA165,123,72,169,0,133,251,165
350 DATA43,133,122,165,44,133,123,32
360 DATA150,93,32,150,93,32,150,93,165
370 DATA45,197,122,165,46,229,123,144
380 DATA47,32,150,93,201,137,240,18
390 DATA201,141,240,14,201,167,240,10
400 DATA201,0,208,227,32,150,93,76,42
410 DATA93,32,115,0,32,107,201,165,3
420 DATA197,20,208,17,165,4,197,21,208
430 DATA11,169,255,133,251,104,133,123
440 DATA104,133,122,96,32,156,93,201
450 DATA44,240,220,201,58,240,161,201
460 DATA137,240,212,201,141,240,208
470 DATA201,0,240,6,32,150,93,76,118
480 DATA93,32,150,93,76,42,93,230,122
490 DATA208,2,230,123,160,0,177,122,96
500 DATA160,0,165,45,197,34,165,46,229
510 DATA35,144,19,177,34,145,36,230,34
520 DATA208,2,230,35,230,36,208,2,230
530 DATA37,76,163,93,56,165,36,233,1
540 DATA133,45,165,37,233,0,133,46,96
550 DATA32,150,93,32,9,201,200,152,24
560 DATA101,252,133,252,176,7,165,252
570 DATA201,249,176,1,96,169,0,133,252
580 DATA96,139,167,137,128,142,143,49685:REMCHECKSUM
```

Listing 1. The loader program.

```
10 DATA 0,255,255,0,0,216,214,8,
207,58,92,0,0,1,1,0,0,0,0,0,1307
20 DATA 0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,205,154,209,6,2,576
30 DATA 126,31,31,31,31,230,15,2
46,48,254,58,56,2,198,7,215,126,
16,242,201,2164
40 DATA 62,2,205,1,22,33,34,207,
221,33,254,207,14,7,221,94,0,221
,35,221,2094
50 DATA 86,0,205,43,207,20,20,20
,20,205,154,209,213,197,94,22,0,
229,235,30,2209
60 DATA 32,1,24,252,205,42,25,1,
156,255,205,42,25,14,246,205,42,
25,125,205,2127
70 DATA 239,21,225,193,209,221,8
6,0,221,35,29,205,154,209,175,86
,6,8,62,48,2432
80 DATA 203,2,48,1,60,215,16,246
,43,13,32,178,42,13,207,1,12,0,9
,30,1871
90 DATA 21,22,30,205,43,207,43,2
9,32,249,30,2,22,1,14,4,33,19,20
7,35,1248
100 DATA 205,43,207,43,20,20,205
,43,207,43,20,20,20,13,32,240,33
,27,207,175,1823
110 DATA 70,30,18,22,11,205,242,
207,20,20,205,242,207,20,20,20,2
03,0,203,0,1965
120 DATA 203,0,205,242,207,20,20
,20,203,0,205,242,207,201,205,15
4,209,62,48,203,2856
130 DATA 0,48,1,60,215,201,6,1,6
,11,10,1,10,11,14,1,14,11,18,1,6
40
140 DATA 62,1,24,2,62,0,50,21,20
7,33,13,207,52,201,19,71,26,254,
233,194,1732
150 DATA 220,209,120,254,253,40,
8,221,110,0,221,102,1,24,6,253,1
10,0,253,102,2507
160 DATA 1,34,13,207,201,237,75,
13,207,62,255,24,5,237,75,13,207
,3,42,15,1926
170 DATA 207,43,112,43,113,34,15
,207,230,56,111,38,0,34,13,207,2
01,237,91,13,2005
180 DATA 207,26,254,251,40,170,2
54,243,40,170,254,16,40,102,254,
233,40,123,254,8,2979
190 DAJA 202,60,209,254,217,202,
84,209,33,97,211,1,5,0,237,177,4
0,115,1,18,2372
200 DATA 0,237,177,202,254,210,1
,2,0,237,177,40,137,1,9,0,237,17
7,202,254,2554
210 DATA 210,1,8,0,237,177,40,16
1,254,237,194,220,209,19,26,254,
70,6,0,40,2363
220 DATA 25,254,86,6,1,40,19,254
,94,6,2,40,13,254,77,202,235,210
,254,69,2141
230 DATA 202,235,210,195,220,209
,120,50,22,207,33,13,207,52,52,2
01,58,27,207,111,2631
240 DATA 229,241,58,30,207,61,50
,30,207,245,225,125,50,27,207,20
3,119,40,66,24,2444
250 DATA 55,33,33,207,17,13,207,
1,2,0,237,176,201,33,27,207,254,
24,40,45,1812
260 DATA 254,56,32,6,203,70,40,2
8,24,35,254,48,32,6,203,70,40,27
,24,16,1468
270 DATA 254,40,32,6,203,118,40,
8,24,15,203,118,40,11,24,0,42,13
,207,35,1433
280 DATA 35,34,13,207,201,42,13,
207,35,78,35,6,0,203,121,40,2,6,
255,9,1542
290 DATA 34,13,207,201,62,2,205,
1,22,6,2,33,27,207,17,35,207,205
,138,209,1833
300 DATA 33,134,209,6,2,195,105,
209,62,2,205,1,22,6,6,33,29,207,
17,37,1520
310 DATA 207,205,138,209,33,122,
209,6,6,94,35,86,35,205,148,209,
62,39,215,16,2279
320 DATA 244,33,13,207,52,201,12
,14,12,4,8,14,8,4,4,14,4,4,16,14
,882
330 DATA 16,4,26,78,235,18,113,3
5,19,16,247,201,62,21,215,62,1,2
15,62,22,1668
340 DATA 215,123,215,122,215,201
,237,75,13,207,42,15,207,43,112,
43,113,34,15,207,2454
350 DATA 237,87,103,58,23,207,11
1,78,35,70,237,67,13,207,201,33,
13,207,53,42,2082
360 DATA 13,207,34,24,207,126,50
,26,207,58,23,207,119,205,89,208
,42,24,207,58,2134
370 DATA 26,207,119,201,205,104,
210,42,13,207,17,10,210,237,176,
34,13,207,221,42,2501
380 DATA 19,207,253,42,17,207,23
7,75,29,207,237,91,31,207,42,27,
207,229,241,42,2647
390 DATA 33,207,237,115,11,207,2
37,123,15,207,0,0,0,0,251,237,11
5,15,207,237,2454
400 DATA 123,11,207,221,34,19,20
7,253,34,17,207,237,67,29,207,23
7,83,31,207,34,2465
410 DATA 33,207,245,225,34,27,20
```

Ian Potts with a small step for your micro, a giant leap for you.

SINGLESTEP ALLOWS YOU to run through machine-code routines one instruction at a time, thus helping you to learn machine-code programming or to de-bug machine-code routines.

The contents of the main registers are displayed in hex, decimal and binary and thus the effect of any operation, especially the logical operations, is easily seen. Register contents, program counter and the flag register can be changed at any stage, with numbers being input in either hex or decimal.

One breakpoint address can be set which allows a long routine or loop to be executed at full speed. Interrupts in all three modes can be simulated. A simple machine-code loader is included which allows machine-code routines to be inserted at the program counter address.

Type in the machine-code loader program which can then be run and saved if correct. Now type New. The Basic can now be typed in, and both Basic and machine code can be saved by entering Go To 9900.

The A, B, C, D, E, H and L registers each have their contents displayed in binary, hex and decimal. the IX, IY, SP — stack pointer — and PC — program counter — register contents are displayed in the boxes in hex. Down the right-hand side of the display is a printout of some of the contents of memory in hex with the current value of the program counter displayed in decimal and pointing to the correct memory location — see figure 1.

Also displayed is the current value of the breakpoint address, BR. The F or Flag register shows the current status of the sign flag, S, zero flag Z, parity or overflow flag P/V, and the carry flag C. If an exchange register instruction is carried out then an apostrophe will be displayed by the registers affected.

First, let's look at the single-letter commands:

S Singlestep, i.e., carry out next machine-code instruction.



SPEC

SINGLE

R Restart program. Clear all registers and reset SP and PC.

Q Quit program and exit to basic. Go To Start will restart the program.

I Simulate an interrupt.

K Continue program at full speed up to breakpoint address.

Here are the commands requiring a value to be input:

P Change program counter to value input

P+ Add value to program counter.

P– Subtract value from program counter.

BR Set breakpoint address to value input.

HEX Inserts hex code starting at program counter address.

F Set flag register to value input — this

```
7,33,0,0,34,10,210,34,12,210,201
,42,9,207,1980
420 DATA 126,50,8,207,54,201,237
,75,13,207,34,13,207,62,205,50,1
0,210,237,67,2273
430 DATA 11,210,205,234,209,58,8
,207,42,13,207,119,201,58,27,207
,230,58,246,0,2550
440 DATA 50,27,207,201,237,91,13
,207,26,254,237,32,14,1,2,0,19,2
6,230,199,2073
450 DATA 254,67,192,1,4,0,201,33
,139,211,1,20,0,237,177,1,2,0,20
0,1,1741
460 DATA 8,0,237,177,1,3,0,200,1
,2,0,237,177,1,1,0,192,19,26,33,
1315
470 DATA 169,211,1,5,0,237,177,1
,4,0,200,1,11,0,237,177,1,3,0,19
2,1627
480 DATA 1,2,0,201,42,13,207,35,
203,72,32,2,35,35,34,13,207,201,
203,72,1610
490 DATA 32,33,203,64,40,17,237,
75,13,207,3,3,3,42,15,207,43,112
,43,113,1505
500 DATA 34,15,207,42,13,207,35,
78,35,70,237,67,13,207,201,42,15
,207,78,35,1838
510 DATA 70,35,34,15,207,237,67,
13,207,201,24,188,24,200,6,0,203
,79,32,8,1850
520 DATA 6,1,203,87,32,2,6,2,203
,71,32,236,230,56,33,27,207,254,
0,32,1720
530 DATA 6,203,118,40,223,24,219
,254,8,32,6,203,118,40,211,24,21
1,254,16,32,2242
540 DATA 6,203,70,40,203,24,199,
254,24,32,6,203,70,40,191,24,191
,254,32,32,2098
550 DATA 6,203,86,40,183,24,179,
254,40,32,6,203,86,40,171,24,171
,254,48,32,2082
560 DATA 6,203,126,40,163,24,159
,203,126,40,155,24,155,24,32,40,
48,56,194,195,2013
570 DATA 196,202,204,205,210,212
,218,220,226,228,234,236,242,244
,250,252,221,253,201,216,4470
580 DATA 208,200,192,232,224,248
,240,199,207,215,223,231,239,247
,255,6,14,16,22,30,3448
590 DATA 38,46,54,62,198,203,206
,211,214,219,222,230,238,246,254
,1,17,33,34,42,2768
600 DATA 49,50,58,221,253,203,33
,34,42,54,9,25,35,41,43,57,249,2
33,225,227,2141
610 DATA 229,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,229
1000> REM 48K LOADER PROGRAM
1010 CLEAR 52999: RESTORE
1020 FOR i=0 TO 600 STEP 10
1030 LET t=0
1040 FOR j=0 TO 19
1050 READ a: POKE (53000+i+i+j),
a
1060 LET t=t+a
1070 NEXT j
1080 READ a: IF a<>t THEN PRINT
FLASH 1;"ERROR AT LINE ";i+10: F
LASH 0: STOP
1090 NEXT i
1100 CLS
1110 SAVE "s/step48K"CODE 53000,
1200
1120 VERIFY ""CODE
```

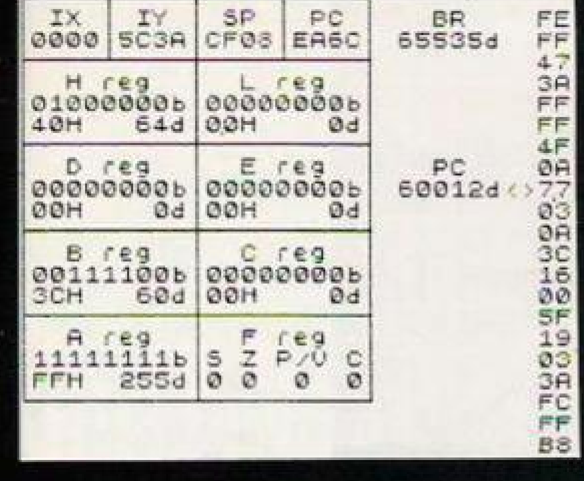


Figure 1. The screen display.

PECTRUM

GLESTEP

must be four digits, each digit either 0 or 1.

Any register letter or register pair will set that register to the value input. If the value input ends in H then the number is interpreted as hex, otherwise it is taken to be decimal. Help will display a summary of the commands.

The stack pointer is initially set to point to the start of the code. The machine code is not relocatable.

On entry, interrupts are enabled and IM1 is selected. Interrupts can be disabled in a program — Singlestep will still work. When IM0 or IM2 is selected in a program then if an interrupt is simulated, the data that the interrupting device supplies must be input. It is assumed that the Spectrum supplies FF when in IM2 or IM3.

Here are some simple programs to illustrate the use of Singlestep. First, load the program. This will produce a display similar to the one shown in figure 1.

Hex	Mnemonic	Comment
06 02	LD B,2	Load B with 2
00	LOOP: NOP	Do nothing
10 FD	DJNZ LOOP	Subtract 1 from B, if result is non-zero jump FD, i.e. −3

To enter this program type Hex followed by Enter. Then type 06020010FD, Enter. The machine code will now be inserted. Note that each byte must be entered as two digits, i.e. 02 for 2 etc. Now type P−, Enter followed by 5 to move back to the start of the program. Press S, Enter and the B register will now contain 2. Press S, Enter and all that will happen is that the program counter increases by 1. Press S, Enter and the program counter will now point to the start of the loop, and the B register will now contain 1.

Continue pressing S followed by Enter until the program counter contains 55005. You have now exited from the loop.

Here is another example program.

Hex	Mnemonic	Comment
06 FF	LD B,255	Load B with 255
00	LOOP: NOP	Do nothing
10 FD	DJNZ LOOP	Loop as before

To change the first example program, type in P 55001. This moves PC to point to the byte to be changed. HEX FF — FF is now inserted. P− 2 means that PC now points to start of the program.

Enter S several times. As you can see it will take a long time to exit from the loop. To execute the code at speed, enter BR 55005, which sets the breakpoint address. K now executes the program at full speed.

Now for the third example:

D9	EXX	Exchange registers
D9	EXX	Exchange registers

To enter this program type HEX D9D9

H 32 — puts 32 decimal into register H

D 32H — puts 32 Hex into register D

P−2 — moves PC to point to start of program

Now singlestep twice, observing the display.

This fourth program simulates an interrupt in Mode 2.

3E 09	LD A,9	
ED 47	LD I,A	Load I with the value 9
ED 5E	IM2	Set interrupt Mode 2

Load this program starting at address 54994 so that it precedes the second program. When you simulate an interrupt you will supply the byte FF so that the interrupt is vectored via the contents of 09FF to FE69. So enter the following:

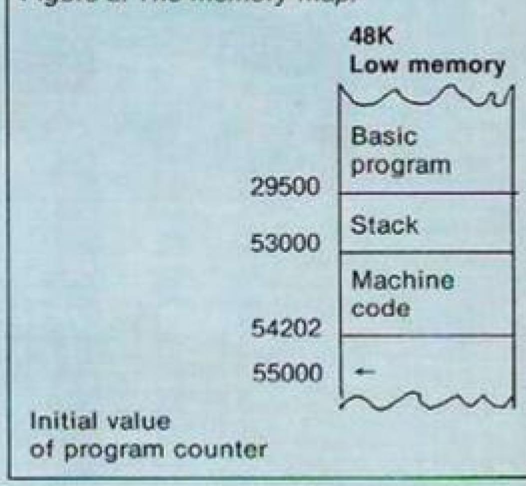
P FE69H	PC now points to FE69
HEX 00ED4D	m/code NOP, RETI is loaded.
P 54994	Return to start of program 4

Singlestep through the loop

Enter S three times to set up interrupt values. Now singlestep some way through the loop of the second program and enter I. In response to the prompt, enter FF. You will now jump to 65129, FE69, you can now continue singlestepping and you will return to the point where the routine was called.

For those who baulk at typing in all the data, I would be prepared to supply copies of the tape at £3 each provided the name and address is clear. Ian Potts, 1 Bramston Close, Oundle, Peterborough PE8 4DP. ■

Figure 2. The memory map.



Listing 2. The Basic program.

```
30 DEF FN H(X)=INT (X/256)
40 DEF FN L(X)=X-256*FN H(X)
50 DEF FN H(H$)=CODE H$-48-7*(H$>="A")
100 LET P=256*PEEK (addr+6)+PEEK (addr+5)
120 PRINT AT 2,22;"     ";AT 2,27-LEN STR$ BR;BR
130 PRINT AT 9,22;"     ";AT 9,27-LEN STR$ P;P
140 RANDOMIZE USR (addr+60): RETURN
3000 GO SUB 100
3100 POKE 23658,0: INPUT "COMMAND? "; LINE Z$
3110 RESTORE 3500: PRINT AT 21,0;"  "
3120 FOR I=1 TO 8: READ A$,A
3130 IF Z$=A$ THEN GO TO A
3140 NEXT I
3150 FOR I=1 TO 18: READ A$,A,C
3160 IF Z$<>A$ THEN NEXT I: PRINT AT 21,0;"WRONG COMMAND": GO TO 3100
3170 INPUT "VALUE? "; LINE Y$
3180 IF Y$="" THEN GO TO 3170
3200 IF Z$="F" THEN GO TO A
3210 IF Y$(LEN Y$)="H" THEN GO TO 5500
3220 FOR I=1 TO LEN Y$: IF Y$(I)<"0" OR Y$(I)>"9" THEN GO TO 6010
3230 NEXT I
3240 LET B=VAL Y$
3250 IF B<0 OR B>65535 THEN GO TO 6000
3260 GO TO A
3300 IF B>255 THEN GO TO 6000
3310 POKE (addr+C),B: GO TO 3000
3340 LET P=P+B: IF P>top THEN LET P=P-B: GO TO 6000
3350 LET B=P: GO TO 3450
3360 LET P=P-B: IF P<0 THEN LET P=P+B: GO TO 6000
3370 LET B=P: GO TO 3450
3380 IF B>top THEN GO TO 6000
3390 LET P=B: GO TO 3450
3400 IF B>top THEN GO TO 6000
3410 LET BR=B
3450 POKE (addr+C),FN L(B): POKE (addr+C+1),FN H(B): GO TO 3000
3500 DATA "",3100,"R",8030,"S",4100,"Q",9999,"I",4500,"K",4110,"HELP",7500,"HEX",5000
3510 DATA "P+",3340,5,"P-",3360,5,"P",3380,5,"BR",3400,1
3520 DATA "F",6500,0,"SP",3450,7,"BC",3450,21,"DE",3450,23,"HL",3450,25,"IX",3450,11,"IY",3450,9
3530 DATA "A",3300,20,"C",3300,21,"B",3300,22,"E",3300,23,"D",3300,24,"L",3300,25,"H",3300,26
4100 RANDOMIZE USR (addr+337): GO TO 3000
4110 RANDOMIZE USR (addr+817): GO TO 3000
4500 IF PEEK (addr+13)=0 THEN GO TO 6040
4503 IF PEEK (addr+14)=1 THEN RANDOMIZE USR (addr+305): GO TO 3000
4505 INPUT "DEVICE DATA (in Hex) "; LINE Z$
4520 LET B=16*FN H(Z$(1))+FN H(Z$(2))
4530 IF B>255 THEN GO TO 6000
4540 POKE (addr+15),B: IF PEEK (addr+14)=0 THEN GO TO 4600
4550 RANDOMIZE USR (addr+666): GO TO 3000
4600 RANDOMIZE USR (addr+695): GO TO 3000
5000 INPUT "CODE? "; LINE Y$
5005 FOR I=1 TO LEN Y$: IF Y$(I)<"0" OR Y$(I)>"F" OR (Y$(I)<"A" AND Y$(I)>"9") THEN GO TO 6020
5007 NEXT I
5010 FOR N=1 TO LEN Y$/2: LET B$=Y$(2*N-1 TO 2*N)
5020 LET B=16*FN H(B$(1))+FN H(B$(2))
5025 IF B>255 THEN GO TO 6000
5030 POKE P,B: LET P=P+1: NEXT N
5040 POKE (addr+6),FN H(P): POKE (addr+5),FN L(P): GO TO 3000
5500 LET B=0: LET Y$=Y$( TO LEN Y$-1)
5505 FOR I=1 TO LEN Y$: IF Y$(I)<"0" OR Y$(I)>"F" OR (Y$(I)<"A" AND Y$(I)>"9") THEN GO TO 6020
5506 NEXT I
5510 FOR I=0 TO LEN Y$-1
5520 LET B=B+FN H(Y$(LEN Y$-I))*16↑I
5530 NEXT I: GO TO 3250
6000 PRINT AT 21,1;"[illegible]": GO TO 3100
6010 PRINT AT 21,1;"[illegible]": GO TO 3100
6020 PRINT AT 21,1;"[illegible]": GO TO 3100
6030 PRINT AT 21,0;"[illegible]": GO TO 3100
6040 PRINT AT 21,1;"[illegible]": GO TO 3100
6500 IF LEN Y$<>4 THEN GO TO 6030
6501 FOR I=1 TO 4: IF Y$(I)<"0" OR Y$(I)>"1" THEN GO TO 6030
6502 NEXT I
6505 LET flag=128*VAL Y$(1)+64*VAL Y$(2)+4*VAL Y$(3)+VAL Y$(4)
6510 POKE (addr+859),flag
6520 RANDOMIZE USR (addr+853): GO TO 3000
7000 CLS : PLOT 4,20: DRAW 0,151: DRAW 159,0: DRAW 0,-151: DRAW -159,0
7010 FOR I=52 TO 148 STEP 32
7020 PLOT 4,I: DRAW 159,0: NEXT I
7040 PLOT 84,20: DRAW 0,151
7050 PLOT 44,148: DRAW 0,23
7060 PLOT 124,148: DRAW 0,23
7070 LET A$=" reg": RESTORE 7120
7090 FOR I=0 TO 6: READ B$,X,Y: PRINT AT X,Y+2;B$;A$
7100 PRINT AT X+1,Y+8;"b": PRINT AT X+2,Y+2;"H": PRINT AT X+2,Y+8;"d": NEXT I
7120 DATA "H",4,1,"L",4,11,"D",8,1,"E",8,11,"B",12,1,"C",12,11,"A",16,1
7130 PRINT AT 16,13;"F";A$
7140 PRINT AT 1,2;"IX";AT 1,7;"IY";AT 1,12;"SP";AT 1,17;"PC"
7150 PRINT AT 1,24;"BR";AT 2,27;"d";AT 8,24;"PC";AT 9,27;"d<>"
7160 PRINT AT 17,11;"S Z P/V C"
7170 RETURN
7500 CLS : PRINT AT 0,9;"[illegible]"
7520 PRINT AT 3,0;"Q  QUIT"'"R  RESTART"'"I  SIMULATE AN INTERRUPT"
7530 PRINT AT 6,0;"S  SINGLESTEP"'"K  CONTINUE AT SPEED"
7540 PRINT AT 8,0;"F  SET FLAGS"'"BR SET BREAKPOINT"'"P  SET P/COUNTER"'"P+ or P-  INCREMENT P/COUNTER"'"HEX  INSERT CODE"
7550 PRINT AT 14,0;"REGISTER LETTER(S) THEN A"'"NUMBER LOADS THE REGISTER"'"NUMBERS ARE DECIMAL UNLESS"'"ENDING IN H"
7560 PRINT AT 20,0;"PRESS ANY KEY TO CONTINUE"
7570 PAUSE 0: GO TO 8000
8000 CLEAR 29500: BORDER 1: PAPER 1: CLS : INK 7: POKE 23658,0: POKE 23609,100: LET START=8100
8010 LET top=65535: LET addr=53000
8020 PRINT AT 8,8; FLASH 1;"DO NOT STOP TAPE";AT 10,10;"LOADING CODE": LOAD ""CODE addr
8030 FOR I=addr TO addr+34: POKE I,0: NEXT I
8040 RESTORE 8200: FOR I=addr+1 TO addr+14: READ A: POKE I,A: NEXT I
8050 LET BR=top: LET P=55000
8100 GO SUB 7000: GO TO 3000
8200 DATA 255,255,0,0,216,214,8,207,50,92,0,0,1,1
9900 CLS : SAVE "S/STEP48K" LINE 8000
9910 SAVE "s/step48K"CODE 53000,1202: PRINT AT 10,0;"[illegible]"
9920 VERIFY "": VERIFY ""CODE
9999 CLS
```



BASIC LIGHTNING
WHITE LIGHTNING
MACHINE LIGHTNING
LIGHTNING STRIKES AGAIN
OASIS SOFTWARE
THREE HIGH POWERED GRAPHICS DEVELOPMENT SYSTEMS FOR THE COMMODORE 64.
Totally dedicated to writing very fast, video games. BASIC Lightning is a is a fully structured extension to the Commodore BASIC which adds a staggering 200 reserved words. It allows up to five tasks to run concurrently (one in foreground and four in background). Most of the commands are dedicated to games writing and the sound and graphics commands are unparalleled. Procedures and PASCAL type structured programming commands are also a feature of BASIC Lightning.
As well as the Commodore's own 8 hardware sprites, BASIC Lightning has its own software sprites. Up to 255 can be defined with user selectable dimensions. These can even be several screens wide. They can be scrolled, spun, mirrored, enlarged or inverted with phenomenal speed and smoothness.
A Sprite Generator Program (written in BASIC Lightning) is also supplied and can be used to design, edit and store all your sprites for use in your main program.
White Lightning is a complete games writing package comprising a high level, Forth based, multi-tasking games writing language, the extended BASIC (see BASIC LIGHTNING) and a powerful sprite Generator Program. Programs can even be written in a combination of Forth and Commodore BASIC and the final program, which will run independently of White Lightning, can be marketed with no restrictions whatsoever.
The Basic Lightning part of the package can be used to experiment quickly and easily before the Forth program is developed.
The speed of White Lightning has to be seen to be believed and a full demo is included. As with Basic Lightning, hardware sprites are supported, together with 255 software sprites which can be scrolled, spun, reflected, enlarged or inverted.
MULTI-TASKING Without doubt the most powerful feature of the Lightning series of languages is the multi-tasking facility. This allows two programs to be run concurrently and makes those smooth landscape scrolls etc. effortless.
The BASIC Lightning Sprite Generator Program is also included in the package.
Commodore 64 Machine Lightning is probably the most advanced games writing utility available on any micro. It comes in 4 parts:
MACRO ASSEMBLER/MONITOR/DISASSEMBLER This is probably the most comprehensive machine code development system available for the Commodore 64 with features too numerous to mention.
BASIC LIGHTNING BASIC Lightning, the multi-tasking BASIC is also provided to facilitate experimentation in preparation for later assembly.
SPRITE GENERATOR Used to develop all the graphics for the final game. The Sprite Generator has numerous functions including enlargement, rotation and reflection.
OBJECT LIBRARY This is Machine Lightning's most powerful feature. 10k of re-entrant code with more than 130 documented entry points. These are the routines that provide all the superfast graphics routines in White and Basic Lightning. They contain virtually every routine you'll ever need to write an Arcade Game and multi-tasking in Machine Lightning is covered in the comprehensive accompanying manual.
AVAILABLE ON TAPE OR DISK
Available from [Boots] and all good software suppliers.
OASIS SOFTWARE
9a Alexandra Parade, Weston-super-Mare, Avon BS23 1QT Telephone: (0934) 419921.
Every product carries a lifetime guarantee.
All prices include comprehensive manuals, VAT and p&p.
(includes manuals and p&p overseas)
24-hour Access Tele-ordering on (0934 419921)
Please send me the following Lightning System Pack(s)
Basic Lightning (tape) £14.95
(disk) £19.95
White Lightning (tape) £19.95
(disk) £29.95
Machine Lightning (tape) £29.95
(disk) £39.95
I enclose my cheque/P.O. for £
Name
Address
Telephone

TELSOFT



A reminder of how to use the Telsoft service.

THE PROGRAMS given here will enable Spectrum and BBC owners to download via *Your Computer's* Telsoft service. First type the hexloader for your machine — figure 1 — and then enter the machine code — figure 2. On the BBC you call the service by entering

CALL &6A00

while Spectrum owners must type

RANDOMIZE USR 60000

So far OE Ltd's Telemod 2 and VTX 5000 modems have been tested with the service, but it also works with a number of other makes.

To find out what is available on Telsoft and how to receive software dial up Colchester (0206) 8068. This audio recorded information line will also advise you which telephone numbers to ring for the 300 and 1200 bit/s services.

When a program you want to download is available, make sure your modem is set up and dial the number appropriate to your modem's speed. As soon as you hear the modem tones switch the modem to line and replace the receiver. Select Option 1 from the menu — Receive. After a block of data is received, you will see "OK" printed if there were no errors, otherwise the program will wait for the blocks to come round again.

When the "Program loaded OK" message appears return to the Telsoft menu and select Option 5. You can now save and run the program.

Figure 1. BBC.

```
10 REM BBC HEX CODE LOADER
15 HIMEM=&69FF
20 CLS:PRINT
30 INPUT " START ADDRESS (Hex)";A$
40 A=EVAL("&"+A$)
50 IF A>&6F87 THEN 280
60 IF A<&6A00 OR A>&6FE7 THEN 20
70 PRINT "A" ";
80 INPUT ":" B$,C$
90 IF LEN(B$) <>16 THEN 50
100 T=0
110 FOR N=0 TO 7
120 X$= MID$(B$,2*N+1,1): GOSUB 300
130 IF E=1 THEN 260
140 X$= MID$(B$,2*N+2,1): GOSUB 300
150 IF E=1 THEN 260
160 B= EVAL("&"+MID$(B$,2*N+1,2))
170 ?A=B:A=A+1:T=T+B
180 NEXT
190 FOR M = 1 TO LEN (C$)
200 X$=MID$(C$,M,1):GOSUB 300
210 IF E =1 THEN A=A-1: GOTO 260
220 NEXT
230 IF T= EVAL("&"+C$) THEN 50
240 PRINT "CHECKSUM ERROR !"
250 A=A-8:GOTO 50
260 PRINT "TYPING ERROR !"
270 A=8*(A DIV 8):GOTO 50
280 *SAVE "DOWNLOAD" 6A00 6F87
290 END
300 E=0:IF ASC(X$)<48 THEN E=1:RETURN
310 IF ASC(X$)<58 THEN RETURN
320 IF ASC(X$)<65 THEN E=1:RETURN
330 IF ASC(X$)>71 THEN E=1
340 RETURN
```

Figure 2. BBC.

```
6A00 :A9C8A0FEA20120F4,4C6
6A08 :FF20616C204B6DC9,38A
6A10 :31F00BC934F0E9C9,4CB
6A18 :35F0064C0C6A4C47,280
6A20 :6AA90C20E3FFA9E0,4B2
6A28 :A000A2FF20F4FFA9,4FD
6A30 :03A20020F4FFA902,363
6A38 :A20220F4FFA9C8A0,4CB
6A40 :FEA20020F4FF60A9,48C
6A48 :0C20E3FF20F76BA9,439
6A50 :15A20120F4FFA000,3E8
6A58 :A9FF8899856FD0FA,587
6A60 :A915A20120F4FF20,394
6A68 :0C6C90F8200C6C80,348
6A70 :F800008477847884,416
6A78 :79047A047B057B20,38D
6A80 :056DA57029F0C980,489
6A88 :D0E2200C6C80D099,468
6A90 :7100C020056DC809,364
6A98 :D0F0A000A5742980,422
6AA0 :D000A51C857610A5,354
6AA8 :1065730577200C6C,289
6AB0 :B00520E3FFC82005,524
6AB8 :6DC010D0F0A57320,435
6AC0 :B26D200C6C80A0C5,3CC
6AC8 :7AF000A95020E3FF,478
6AD0 :20E7FF4C676A200C,34F
6AD8 :6C808CC57BF000A9,40C
6AE0 :5020E3FF20E7FF4C,4AC
6AE8 :676AA573A0B9056F,43E
6AF0 :D00E20E7FFA90820,388
6AF8 :E3FF4C836B4C676A,469
6B00 :A000847A047B200C,2C9
6B08 :6C80F2917620D56D,477
6B10 :C8C475D0F1200C6C,45A
6B18 :B0E3C57AF00BA950,4CE
6B20 :20E3FF20E7FF4C67,48B
6B28 :6A200C6C80CFC57B,3C1
6B30 :F00BA95020E3FF4C,447
6B38 :676A20C76B20E7FF,429
6B40 :A573A0A90099856F,3F6
6B48 :A58AC902D0502007,419
6B50 :6DA9C820056CA955,3B8
6B58 :20306EA95020556C,298
6B60 :A915A20120F4FF20,394
6B68 :F76BA471C888B985,5B5
6B70 :6FD05198D0F7A90C,4A4
6B78 :20E3FF20A56DA95A,437
6B80 :0582A96F8583A907,3D7
6B88 :20006E20F86D20A8,2DB
6B90 :6DA90520006EA908,2DA
6B98 :20E3FF20F86DA9D2,502
6BA0 :20E0FF4C0C6AA955,38F
6BA8 :20306EA99620556C,2DE
6BB0 :4C6A6BA475200C6C,2D2
6BB8 :88D0FA200C6C200C,316
6BC0 :6C4C486B4C6C6AA9,336
6BC8 :2020E3FFA96F20E3,430
6BD0 :FFA9AB20E3FFA0A9,51E
6BD8 :9CA000A6BAE0020D,41E
6BE0 :0BA21320F4FFA212,387
6BE8 :20F4FF60A29720F4,4C0
6BF0 :FFA29620F4FF60A9,553
6BF8 :9CA000A6BAE0020D,41E
6C00 :E8A25320F4FFA252,4E7
6C08 :20F4FF6018A99605,44F
6C10 :0F847EA991A20820,38D
6C18 :F4FFB009C007D005,44B
6C20 :60604C0C6AA47E20,2D4
6C28 :496CC60FD0023B60,374
6C30 :A58AC90200062079,369
6C38 :6EB0ECA0A991A201,447
6C40 :20F4FF90A47EB0DF,55C
6C48 :6080A202CAE0CAD0,5AA
6C50 :F8AA60A900847EA0,458
6C58 :20496C80D0FAA47E,449
6C60 :60A9E8A000A2FF20,452
6C68 :F4FFA900850AA90C,460
6C70 :20E3FF20A06DA904,3C4
6C78 :20006EA9098582A9,2F0
6C80 :6F8583A90020F86D,39C
6C88 :20156DC941F00BC9,370
6C90 :42F011C943F0174C,3A2
6C98 :616CA900A20120F4,335
6CA0 :FF4C876CA900A203,3C4
6CA8 :20F4FF4C876CA900,433
6CB0 :A20420F4FFE60AA5,4CE
6CB8 :7C20E3FFA47E20A8,46B
6CC0 :6DA90820006E20F8,2C4
6CC8 :6DA01720156DC941,290
6CD0 :F00BC942F011C943,413
6CD8 :F0174C616CA907A2,372
6CE0 :0120F4FF4C076DA9,37F
6CE8 :07A20320F4FF4C09,314
6CF0 :6DA907A20420F4FF,30A
6CF8 :E60AA50AC902D009,443
6D00 :A90DA000A20120F4,3EB
6D08 :FFA57C20E3FF2053,495
6D10 :6C20536CA020A86D,2E3
6D18 :A90520006E20F86D,2C1
6D20 :20A86DA90520006E,274
6D28 :20F86D20A86DA905,36B
6D30 :20006E20F86D20A8,2DE
6D38 :6D847EA90320006E,2A9
6D40 :A9D220E0FF057C60,4DD
6D48 :A90C20E3FF20A86D,3FF
6D50 :A90720006EA99A85,306
6D58 :82A96E8583A00020,361
6D60 :F86D20A86D20996D,3C8
6D68 :20996D20996D20A8,317
6D70 :6DA90A20006E20F8,2C6
6D78 :6D20A86DA9022000,270
6D80 :6E20F86D20A86D20,348
6D88 :E7FFA90220E0FF40,5A0
6D90 :A90FA20020F4FF60,3D5
6D98 :6DA90820006E20F8,287
6DA0 :6D20A86D6020A86D,33A
6DA8 :20A86D20E7FF20E7,445
6DB0 :FF604029F06A6A6A,3FE
6DB8 :6A0930C93A300318,1F1
6DC0 :690720E3FF60290F,312
6DC8 :0930C93A30031069,1F0
6DD0 :0720E3FF60457885,3AE
6DD8 :7BA200A57B2A900C,3B8
6DE0 :A57B4900857BA57A,390
6DE8 :4910857A267A267B,299
6DF0 :CAD0E86020E3FFC8,5AC
6DF8 :B102C90DD0F6C060,4F7
6E00 :867DAAA92020E3FF,478
6E08 :CAD0FAA67D60C97F,55F
6E10 :D000A90A20F4FFE0,4FD
6E18 :00F012A97FC92080,3C3
6E20 :0EC900F00AC90DF0,3A1
6E28 :06C907F002A90060,2D1
6E30 :857C867D847EA991,440
6E38 :A20120F4FFA903A2,404
6E40 :0720F4FFA996A200,403
6E48 :20F4FF982902F0F4,40A
6E50 :A47CA997A20720F4,41F
6E58 :FFA903A20420F4FF,464
6E60 :A991A20020F4FFB0,49F
6E68 :09C007D005A6A84C,2C1
6E70 :0C6AA57CA67DA47E,3DC
6E78 :60847E867DA99AA2,446
6E80 :0020F4FF982901F0,3CD
6E88 :08A996A20920F4FF,400
6E90 :9810900130A67DA4,340
6E98 :7E60444F574E4C4F,2B1
6EA0 :4144494E47204D45,215
6EA8 :4E550D3120202052,193
6EB0 :4543454956450D34,1F2
6EB8 :2020205345542042,1AE
6EC0 :6175642052617465,2E6
6EC8 :0D35202020455849,188
6ED0 :5420544F20424153,20D
6ED8 :49430D454E544552,217
6EE0 :204E554D4245520D,1F6
6EE8 :2020555345204354,1EC
6EF0 :524C2D4720544F20,1F5
6EF8 :52455455524E2054,254
6F00 :4F204D454E552029,1ED
6F08 :0D53455420545241,200
6F10 :4E534D4954204261,24E
6F18 :756420526174650D,292
6F20 :4120202037352042,16F
6F28 :6175640D42202033,1FC
6F30 :303020426175640D,209
6F38 :4320313230302042,188
6F40 :6175640D53455420,253
6F48 :5245434549564520,223
6F50 :4261756420526174,2C3
6F58 :650D50524F475241,23D
6F60 :4D20204C4F414445,1F2
6F68 :4420206F60005052,20D
6F70 :45535320414E5920,213
6F78 :4B455920464F5220,210
6F80 :4D454E550D202020,1A2
```

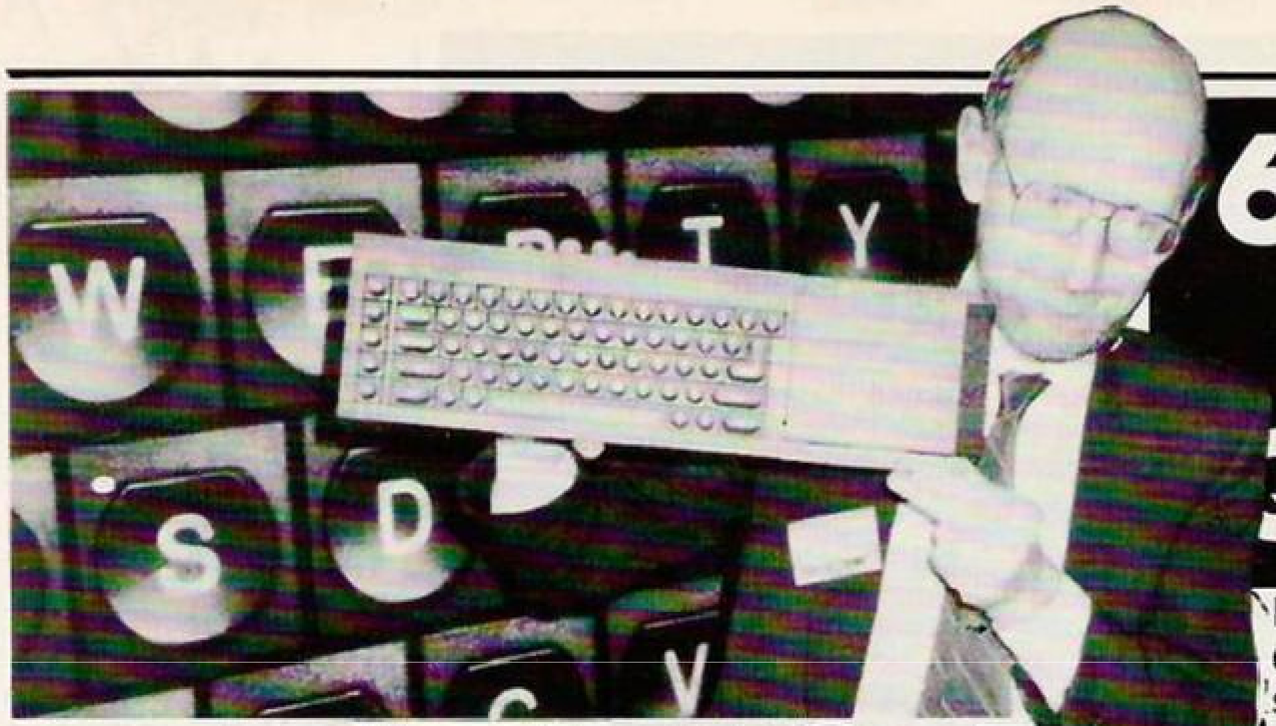
Figure 1. Spectrum.

```
5 REM SPECTRUM 48k          fig 1
10 REM Hex Code Loader
15 CLEAR 59000
20 POKE 23658,0: CLS : PRINT
30 INPUT "Start Address ";a
50 IF a>61135 THEN GO TO 280
60 IF a<60000 THEN GO TO 20
70 PRINT a;
80 INPUT " :";b$
85 IF b$="END" THEN GO TO 280
90 IF LEN b$<>20 THEN GO TO 260
100 LET t=a-256*INT (a/256)
110 FOR n=0 TO 7
120 LET x$=b$(2*n+1 TO 2*n+1)
125 GO SUB 300: LET y=x
130 IF e=1 THEN GO TO 260
140 LET x$=b$(2*n+2 TO 2*n+2)
145 GO SUB 300: LET y=y*16+x
150 IF e=1 THEN GO TO 260
170 POKE a,y: LET a=a+1
180 LET t=t+y: NEXT n: LET y=0
190 FOR m=1 TO 3
200 LET x$=b$(17+m TO 17+m)
205 GO SUB 300: LET y=y*16+x
210 IF e=1 THEN LET a=a-1: GO TO 260
220 NEXT m
230 IF t=y THEN PRINT ":";b$: GO TO 50
240 PRINT "Checksum Error"
250 LET a=a-8: GO TO 50
260 PRINT "Typing Error"
270 LET a=8*INT (a/8): GO TO 50
280 SAVE "download"CODE 60000,1136
290 POKE 23658,0: STOP
300 LET e=0: LET x=CODE x$-48-7*(x$>"9")
310 IF x<0 OR x>15 THEN LET e=1
320 RETURN
```

Figure 2. Spectrum.

```
60000 :CD15EDC307ECCD15,4C7
60008 :EDCD36ECCD66EDCD,631
60016 :50EDCD47EE30FBCD,5AF
60024 :0BEDFE31CA99EAFE,69A
60032 :35CA87EAC366EACD,5D0
60040 :15ED3EFACD84EDCD,5FD
60048 :15EDC9CD15EDC366,553
60056 :EACD66ED11E5EE06,58C
60064 :803FFF121310FCCD,45C
60072 :15EE11CDEE06173E,3D2
60080 :20121310FCCD1FED,3DA
60088 :30F0CD50EDCD1FED,5BB
60096 :30FD326FEF3E0032,3E9
60104 :6CEF326FEF3270EF,544
60112 :3A65EFCDEBEC3A65,5A1
60120 :EFE6F0FE0020DE0E,627
60128 :091165EFCD1FEDDA,501
60136 :AAEA1312CDEBEC8D,552
60144 :70F23A69EFE60020,51A
60152 :00214B5C3A6BEF0E,36C
60160 :004709226BEF0E00,1DA
60168 :11CDEECD1FED309A,47F
60176 :12130CCDEBEC79FE,45C
60184 :1020F03A6BEFCD81,447
60192 :ECCD1FEDDAAAEA21,574
60200 :6FEFBE2B133E5011,326
60208 :E0EE12133E0012CD,340
60216 :D7ECCDAAECC3AAEA,681
60224 :CD1FEDDAAAEA2170,518
60232 :EFBE20E121E5EE3A,524
60240 :6BEF056F3E000C67,3CC
60248 :3E00BE2000CD07EC,40F
60256 :3E2132085CC3FFEB,402
60264 :3E00326FEF3270EF,3C7
60272 :ED5B6BEF0E00CD1F,40C
60280 :EDDAAAEA3273EFCD,634
60288 :EDCC3A6BEFFE0020,506
60296 :053A69EFE6003A73,432
60304 :EF12130C3A6AEF89,3FC
60312 :200CCD1FEDDAAAEA,50B
60320 :216FEFBEC22DEBCD,504
60328 :1FEDDAAAEA2170EF,5A7
60336 :BEC22DEB3E6F32E2,509
60344 :EE3E6B32E53EE300,490
60352 :32E4EE3A005CFE21,501
60360 :2003CD1BEECD07EC,553
60368 :CDA6EC21E5EE3A60,5C5
60376 :EF056F3E000C6736,422
60384 :0021E5EEFD48A6EF,561
60392 :0C7EFE00C20DEA23,4FC
60400 :0D20F63EFACD84ED,589
60408 :3ED103FEC366EACD,5E8
60416 :1FED30FBC3BDEAF3,594
60424 :0603AFD3FF10FB3E,3DB
60432 :40D3FF3EFFD3FF3E,56F
60440 :31D3FF3E0F320F5C,305
60448 :320D5C32405C3E01,250
60456 :D3FE21915CCB9EFB,56B
60464 :CD50EDC366EACD15,52F
60472 :EECD66ED3E01D3FE,556
60480 :2165EE3E08CD0FEC,442
60488 :CD04ECCDA3ECCDA6,654
60496 :EC23CD77ECCD77EC,5BF
60504 :CDA0ECCD84ECCDA6,661
60512 :EC233E00CD0FECCD,4CF
60520 :B4EC23CDA3EC3E03,49B
60528 :CD0FECCD04ECC9CD,60B
60536 :A3EC3E07CD0FECCD,561
60544 :84EC23C97EFE00C0,520
60552 :CDBBED23C384ECFE,651
60560 :00C0F53E20CDBBED,520
60568 :F13DC30FECCDA0EC,65D
60576 :CDA3ECCDA6EC3E0D,5A6
60584 :CDBBED3EFF320C5C,574
60592 :C9F513E6F0CB1FCB,60C
60600 :1FCB1FCB1FF630FE,4CF
60608 :3AFAC6ECC6071213,498
60616 :F1EA0FFA30FE3A0A,606
60624 :B4ECC6071213C921,46C
60632 :CDEEE5CD84ECE106,69C
60640 :173E20772310FC3E,339
60648 :0077C9E32170EFAE,538
60656 :77C50E007EA7CB17,449
60664 :3000D7EEE0773A6F,3C9
60672 :EFEE10326FEF3721,3D5
60680 :6FEFCB1623CB160D,358
60688 :20E2C1E1C9110501,394
60696 :216A00CDB503C93E,32F
60704 :963277EFCD47EDDA,529
60712 :2EEDF1C366EACDAC,5C0
60720 :ED7BE601F602D3FE,548
60728 :AF2177EF35200237,2FC
60736 :C9CD77ED3BEBC9CD,5F0
60744 :47EEDBFECDBC37C9,5E6
60752 :F5DBFF3A81EFFE00,5C7
60760 :20043E3610023E37,177
60768 :D3FFDB7FF1C9F53A,675
60776 :81EFFE0020043E36,376
60784 :D3FFCD50EDF1C93E,644
60792 :00DDFFE60037CDDD,59[illegible]
60800 :FFE60237CBDDFFE6,626
60808 :302004DB7FA7C93A,3E0
60816 :81EFFE0020043E37,397
60824 :10023E3ED3FFDB7F,452
60832 :C9CD47ED3A73EFDD,5DE
60840 :F1C366EAF53ED23D,5EE
60848 :20FDF1C9CDACED3D,62A
60856 :20FAC9C5D5E5F5FE,70D
60864 :7F2813CDF9EDFE0C,537
60872 :200CFE0D20053E20,20A
60880 :D73E0DD718103E20,34F
60888 :D73E0DD73A0D5CFE,4ED
60896 :2120833E0D073E5F,2E6
60904 :D73E08D7F1E1D1C1,640
60912 :C9FE07201CFECB20,4F3
60920 :19FE0C2014FE0020,3B8
60928 :18FE0D200CFE20FA,367
60936 :0FEEFE80FA11EE3E,4BA
60944 :00C93E07C9CD6B0D,32C
60952 :3E02CD01163E1032,1C4
60960 :095CC93E121601CD,302
60968 :1122C93E121600CD,257
60976 :1122C9C5D5E53E02,3EB
60984 :CD0116CD47EE30FB,451
60992 :CD5CEEE1D1C1C93A,5CD
61000 :385CCB6F200C3A00,20F
61008 :5CF5213B5CCBAEF1,4C3
61016 :A7C937C9F5111000,3DE
61024 :CD10EDF1C9444F57,4D6
61032 :4E4C4F4144494E47,2B4
61040 :204D454E55003120,216
61048 :2020524543454956,276
61056 :4500352020204578,217
61064 :697420746F204261,32B
61072 :7369630020202073,2DF
61080 :652053796D626F6C,393
61088 :2053686966742020,388
61096 :204720746F205265,2C9
61104 :7475726E20746F20,39C
61112 :4D454E5520290045,27B
61120 :4E544552204E554D,309
61128 :4245522000000000,1C1
```


HOT FROM
SEGA®
THE ARCADE WINNERS
released *NOW*
SPY HUNTER
THE OFFICIAL HOME VERSION OF BALLY MIDWAY'S ARCADE HIT
• You control the turbo charged race car/hydro spy boat.
• You control a deadly arsenal of missiles, machine guns, oil slicks and smoke screens.
• Sophisticated spy-challenging graphics.
U.S. Gold is stocked by
John Menzies, WILDINGS
WOOLWORTH, WHSMITH,
Rumbelows
and Spectrum Shops
Bally/MIDWAY
SPY HUNTER
OFFICIAL ARCADE GAME
NEW!
OFFICIAL SEGA ARCADE
Up'n Down
NEW!
BUCK ROGERS
Congo Bongo
ZAXXON
Bally/MIDWAY
TAPPER
NEW!
64 COMMODORE CASSETTE 9·95
DISK 12·95
SPECTRUM CASSETTE 7·95
U.S. GOLD
All American Software
Exclusive manufacturers for SEGA and Bally MIDWAY
For release schedule and trade details contact:
Centre Soft, Unit 10, The Parkway Industrial Centre, Heneage Street, Birmingham B7 4LY. Telephone: 021-359 3020. Telex: 337268.



68008 QL

TUTORIAL

Phil Holliday concludes his short course on assembly language.

THIS ARTICLE on QL assembly language rounds off the series by presenting a long program example. It is machine code monitor which allows you to examine and alter registers and memory, and test another program by stepping through it one instruction at a time. You can even single step through the routines in Rom.

The program is position independent and is loaded by typing 'EXEC_W MDV1_DEBUG_EXC' assuming that it is on a cartridge in Microdrive 1. Note that the QL should be in monitor mode rather than TV mode. (I find the resolution perfectly satisfactory when used with a TV however). All numbers are entered and printed in hex, and all commands and numbers can be entered in either upper or lower case. Spaces between command/parameter fields in general are not necessary and will be ignored. The program accepts the following commands:

Display registers — R or r. This command takes no arguments and displays the contents of all address and data registers, the program counter and the status registers. The status register is displayed in binary.

Change address register — A or a. This command takes the form An=x where n is the register number, between 0 and 7, and x is the new contentsd of the register in hex. So the command A7=40000 will set the stack pointer (i.e. address reg 7) to point to top of memory in the QL, i.e. $40000.

Change data register — D or d. The syntax for this command is identical to the change address register command above.

Change program counter — P or p. The syntax for this command is of the form P=x where x is the new contents of the program counter e.g. P=34000 sets the program counter to $34000.

Change flags — F or f. This command toggles the stage of one of the five flags and takes the form FX,FN,FZ,FV and FC to change the extend, negative, zero, overvlow or carry flag as appropriate.

Change/Display memory — M or m. This command can have an optional suffix of .b, .w or .l to indicate the size of the memory location to be changed, or the size of the location or block of memory to be displayed. Size word is assumed when no size specifier is given. Only one mrmoey location can be changed at a time. The command is in the form M(.s) a=x where .s is the optional size specifier, a is the address of the memory location and x is the new contents of that location. So the command M.b 26000=FF will change the byte at $26000 to $FF. The commands to display a single memory location or a block of memory are of the form M(.s) a and M(.s) a,b where a and b are start and end addresses. If a is greater than b then the block from b to a will be displayed. A dump of a block of memory can be interrupted by hitting any key.

Load an executable program — L or l. This enables a program to be loaded into the transient program area. It takes the form L MDV1_PROG_TEST. After it has been loaded the program counter will be loaded with the start address of the program and A7 (the stack pointer) will be loaded with the top address of the associated data area. To examine these type R. If the program is loaded correctly then 'ok . . .' is printed to the screen, else one of the normal QDOS error messages will be printed. To try it you can even type L MDV1_DEBUG_EXC and single step through the debug program!

Single step — S or s. This command takes no parameters. Every time it is used the a single instruction will be executed. Using the R command enables a display of the new state of the processor after executing the command.

Quit — Q or q. Takes no arguments but just returns you to basic.

For those of you who do not wish to type it all out, or for those of you who do not have an assembler, I will be happy to provide a copy of both the source and executable program on your supplied Microdrive cartridge. Just send three pounds to cover my time, postage and packing to P.A. Holliday, 44 Lennard Road, London SE20 7LX.

```
LOC     OBJECT      SOURCE STATEMENT

                    *Debug for QL.* Feb 1985
                    *Assembled with METACOMCO assembler

=FFFF               err_nc     equ -1
=FFFD               err_om     equ -3

=0000               black      equ 0
=0001               blue       equ 1
=0002               red        equ 2
=0003               magenta    equ 3
=0004               green      equ 4
=0005               cyan       equ 5
=0006               yellow     equ 6
=0007               white      equ 7

                    QDOS       macro
                               moveq #\1&$FF,d0
                               trap #\1/$100
                               tst.l d0
                               endm

                    VECTOR     macro
                               movea.w \1,\2
                               jsr (\2)
                               tst.l d0
                               endm

                    *traps

=0105               mt_frjob   equ $105
=0106               mt_free    equ $106
=0107               mt_trapv   equ $107
=0118               mt_alchp   equ $118

=0201               io_open    equ $201
=0202               io_close   equ $202

=0301               io_fbyte   equ $301
=0302               io_fline   equ $302
=0305               io_sbyte   equ $305
=0307               io_sstrg   equ $307
=030C               sd_bordr   equ $30C
=030E               sd_cure    equ $30E
=0310               sd_pos     equ $310
=0320               sd_clear   equ $320
=0327               sd_setpa   equ $327
=0328               sd_setst   equ $328
=0329               sd_setin   equ $329
=032D               sd_setsz   equ $32D
=0347               fs_headr   equ $347
=0348               fs_load    equ $348

                    *vectors

=00CC               ut_err     equ $CC
=00D0               ut_mtext   equ $D0
=00F6               cn_itohw   equ $F6
=00FA               cn_itohb   equ $FA
=00FE               cn_itohl   equ $FE

=FFFF               this_job   equ -1

                    *open console

0000                debug
0000    72FF                   moveq #this_job,d1
0002    7602                   moveq #2,d3
0004    41FA 0016              lea devname,a0
                               QDOS io_open
0008    7001                           moveq #io_open&$FF,d0
000A    4E42                           trap #io_open/$100
000C    4A80                           tst.l d0
000E    6600 00xx              bne error

                    *save channel id
0012    41FA 000C              lea channel,a0
0016    2088                   move.l a0,(a0)
0018    6008                   bra.s debug1

001A                devname
001A    0004                   dc.w 4
001C    434F 4E5F              dc.b 'CON_'
0020                channel
0020    =0004                  ds.l 1

                    *red border 2 pixels wide
0024                debug1
0024    123C 0002              move.b #red,d1
0028    343C 0002              move.w #2,d2
002C    367C FFFF              movea.w #this_job,a3
                               QDOS sd_bordr
0030    700C                           moveq #sd_bordr&$FF,d0
0032    4E43                           trap #sd_bordr/$100
0034    4A80                           tst.l d0
0036    6600 00xx              bne error

                    *char size 5 by 9 pixels
003A    323C 0000              move.w #0,d1
```

(listing continued on next page)

(listing continued from previous page)

```
0046'  3401            move.w d1,d2
0048'  363C FFFF       move.w #-1,d3
                       QDOS sd_setsz
004C'  7020                    moveq #sd_setsz&$FF,d0
004E'  4E43                    trap #sd_setsz/$100
0050'  4A80                    tst.l d0
0052'  6600 0074       bne error

                   *black paper,black strip,green ink
0056'  123C 0000       move.b #black,d1
                       QDOS sd_setpa
005A'  7027                    moveq #sd_setpa&$FF,d0
005C'  4E43                    trap #sd_setpa/$100
005E'  4A80                    tst.l d0
0060'  6600 006C       bne error
                       QDOS sd_setst
0064'  7028                    moveq #sd_setst&$FF,d0
0066'  4E43                    trap #sd_setst/$100
0068'  4A80                    tst.l d0
006A'  6600 0062       bne error
006E'  123C 0004       move.b #green,d1
                       QDOS sd_setin
0072'  7029                    moveq #sd_setin&$FF,d0
0074'  4E43                    trap #sd_setin/$100
0076'  4A80                    tst.l d0
0078'  6600 0054       bne error

                   *clear window,cursor at top lh corner
                       QDOS sd_clear
007C'  7020                    moveq #sd_clear&$FF,d0
007E'  4E43                    trap #sd_clear/$100
0080'  4A80                    tst.l d0
0082'  6600 0040       bne error
0086'  323C 0000       move.w #0,d1
008A'  3401            move.w d1,d2
                       QDOS sd_pos
008C'  7010                    moveq #sd_pos&$FF,d0
008E'  4E43                    trap #sd_pos/$100
0090'  4A80                    tst.l d0
0092'  6600 003A       bne error

                   *write signon msg to window,enable cursor
0096'  343C 001C       move.w #msg1_len,d2
009A'  363C FFFF       move.w #-1,d3
009E'  43FA 0044       lea msg1,a1
                       QDOS io_sstrg
00A2'  7007                    moveq #io_sstrg&$FF,d0
00A4'  4E43                    trap #io_sstrg/$100
00A6'  4A80                    tst.l d0
00A8'  6600 0024       bne error
                       QDOS sd_cure
00AC'  700E                    moveq #sd_cure&$FF,d0
00AE'  4E43                    trap #sd_cure/$100
00B0'  4A80                    tst.l d0
00B2'  6600 001A       bne error

                   *set up trap vectors
                   *first copy normal traps
00B6'  41FA 0048       lea trap_tbl,a0
00BA'  43F8 008C       lea $8C,a1
00BE'  22D8            move.l (a0)+,(a1)+
00C0'  22D8            move.l (a0)+,(a1)+
00C2'  22D8            move.l (a0)+,(a1)+
00C4'  22D8            move.l (a0)+,(a1)+
00C6'  22D8            move.l (a0)+,(a1)+
00C8'  22D8            move.l (a0)+,(a1)+

                   *set up trace trap for single step
00CA'  43FA 03F6       lea trace_ret,a1
00CE'  21C9            move.l a1,(a0)+

                   *copy level 7 int trap vector
00D0'  43F8 007C       lea $7C,a1
00D4'  2091            move.l (a1),(a0)
00D6'  72FF            moveq #this_job,d1
00D8'  43FA 0026       lea trap_tbl,a1
                       QDOS mt_trapv
00DC'  7007                    moveq #mt_trapv&$FF,d0
00DE'  4E41                    trap #mt_trapv/$100
00E0'  4A80                    tst.l d0
00E2'  6060            bra.s debug2
00E4'              msg1
00E4'  514C 2044 6562 7567     dc.b 'QL Debug ver 1.1 c FAH 1985',$0A
       2076 6572 2031 2E31
       2063 2046 4148 2031
       3938 350A
0100'              msg1end
       =001C       msg1_len  equ msg1end-msg1
0100'                  cnop 0,4
0100'  =004C       trap_tbl  ds.l 19

                   *start of main loop
                   *read in line from kbd,echoed to console
014C'              debug2
014C'  343C 0050       move.w #buf_len,d2
0150'  363C FFFF       move.w #-1,d3
0154'  41FA FEB2       lea channel,a0
0158'  2050            move.l (a0),a0
015A'  43FA 0096       lea buffer,a1
                       QDOS io_fline
015E'  7002                    moveq #io_fline&$FF,d0
0160'  4E43                    trap #io_fline/$100
0162'  4A80                    tst.l d0
0164'  6600 07A0       bne error

                   *get first char of line in d4
0168'  41FA 0090       lea buffer,a0
016C'  1818            move.b (a0)+,d4

                   *quit debug on Q or q
016E'  0C3C 0051       cmp.b #'Q',d4
0172'  6700 0753       beq quit
0176'  0C3C 0071       cmp.b #'q',d4
017A'  6700 0740       beq quit

                   *display registers on R or r
017E'  0C3C 0052       cmp.b #'R',d4
0182'  6700 052E       beq print_reg
0186'  0C3C 0072       cmp.b #'r',d4
018A'  6700 0526       beq print_reg

                   *change address reg on A or a
018E'  0C3C 0041       cmp.b #'A',d4
0192'  6700 00B0       beq chng_addr_reg
0196'  0C3C 0061       cmp.b #'a',d4
019A'  6700 00A8       beq chng_addr_reg

                   *change data reg on D or d
019E'  0C3C 0044       cmp.b #'D',d4
01A2'  6700 00B0       beq chng_data_reg
01A6'  0C3C 0064       cmp.b #'d',d4
01AA'  6700 00A8       beq chng_data_reg

                   *change pc on P or p
01AE'  0C3C 0050       cmp.b #'P',d4
01B2'  6700 00B2       beq chng_pc
01B6'  0C3C 0070       cmp.b #'p',d4
01BA'  6700 00AA       beq chng_pc

                   *load exec file on L or l
01BE'  0C3C 004C       cmp.b #'L',d4
01C2'  6700 01AA       beq do_load
01C6'  0C3C 006C       cmp.b #'l',d4
01CA'  6700 01A2       beq do_load

                   *single step on S or s
01CE'  0C3C 0053       cmp.b #'S',d4
01D2'  6700 0270       beq single_step
01D6'  0C3C 0073       cmp.b #'s',d4
01DA'  6700 0268       beq single_step

                   *change flags on F or f
01DE'  0C3C 0046       cmp.b #'F',d4
01E2'  6700 0312       beq flags
01E6'  0C3C 0066       cmp.b #'f',d4
01EA'  6700 030A       beq flags

                   *change/display memory on M or m
01EE'  0C3C 004D       cmp.b #'M',d4
01F2'  6700 037A       beq do_mem
01F6'  0C3C 006D       cmp.b #'m',d4
01FA'  6700 0372       beq do_mem

                   *else loop if non of these
01FE'  6000 FF4C       bra debug2

       =0050       buf_len   equ 80
01FA'  =0050       buffer    ds.b buf_len

                   *change address reg routine
                   *pass reg blk addr in a1
024A'              chng_addr_reg
024A'  43FA 0568       lea a_reg_blk,a1
024E'  6118            bsr.s chng_reg
0250'  6700 FEF2       beq debug2

                   *change data reg routine
                   *pass reg blk addr in a1
0254'              chng_data_reg
0254'  43FA 0576       lea d_reg_blk,a1
0258'  6108            bsr.s chng_reg
025A'  6700 FEE8       beq debug2

                   *change prog counter routine
                   chng_pc
025E'  43FA 05AC       lea pc,a1
0262'  6108            bsr.s chng_reg_4
0264'  6000 FEDE       bra debug2

                   *change reg sub routine
                   *get reg no and ret if out of range
0268'              chng_reg
0268'  611E            bsr.s get_reg_no
026A'  6702            beq.s chng_reg_1
026C'  4E75            rts

                   *compute address of reg in reg blk
026E'              chng_reg_1
026E'  E584            asl #2,d4
0270'  D3C4            adda.l d4,a1

                   *ret if '=' not found
0272'              chng_reg_4
0272'  6140            bsr.s equals
0274'  6702            beq.s chng_reg_2
0276'  4E75            rts

                   *else get new contents of reg in hex in d5
                   *and update reg value in blk
0278'              chng_reg_2
0278'  6100 0058       bsr get_long
027C'  6702            beq.s chng_reg_3
027E'  4E75            rts

                   *and ret with no error reported in d0
0280'              chng_reg_3
0280'  2285            move.l d5,(a1)
0282'  7000            moveq #0,d0
0284'  4E75            rts

                   *get no between 0 and 7
                   *skip white space and ret if eol found
0286'              get_reg_no
0286'  6100 0020       bsr blanks
028A'  6702            beq.s get_reg_no_1
028C'  4E75            rts

                   *get next char,convert to digit in d4
028E'              get_reg_no_1
028E'  7800            moveq #0,d4
0290'  1818            move.b (a0)+,d4
0292'  983C 0030       sub.b #'0',d4
0296'  6D0A            blt.s get_reg_no_2
0298'  B83C 0007       cmp.b #7,d4
029C'  6204            bhi.s get_reg_no_2
029E'  7000            moveq #0,d0
02A0'  4E75            rts

                   *if not 0 to 7 report error in d0
02A2'              get_reg_no_2
02A2'  70FF            moveq #-1,d0
02A4'  4E75            rts

                   *skip white space routine
                   *note entry point is at blanks
                   *incr char pointer
02A6'              blanks1
02A6'  5248            addq #1,a0

                   *routine entry point
                   *test char at (a0)
02A8'  0C10 0020   blanks    cmpi.b #' ',(a0)

                   *loop till non space char found
02AC'  67F8            beq.s blanks1

                   *if char is not eol exit with d0=0
02AE'  0C10 000A       cmpi.b #$A,(a0)
02B2'  6704            beq.s blanks2
02B4'  7000            moveq #0,d0
02B6'  4E75            rts

                   *else report eol  found
02B8'              blanks2
02B8'  70FF            moveq #-1,d0
02BA'  4E75            rts

                   *routine finds '=' char
                   *returns with d0=0 if '=' found
                   *else returns with d0=-1
02BC'              equals
02BC'  6100 FFEA       bsr blanks
02C0'  6702            beq.s equals_1
02C2'  4E75            rts
02C4'              equals_1
02C4'  0C18 003D       cmpi.b #'=',(a0)+
02C8'  6604            bne.s equals_2
02CA'  7000            moveq #0,d0
02CC'  4E75            rts
02CE'              equals_2
02CE'  70FF            moveq #-1,d0
02D0'  4E75            rts

                   *routine converts ascii in hex to long word in
                    d5
                   *skip blanks,ret if next char is eol
02D2'              get_long
02D2'  6100 FFD4       bsr blanks
02D6'  6702            beq.s get_long_1
02D8'  4E75            rts

                   *else clear d5 prior to conversion
02DA'              get_long_1
02DA'  7A00            moveq #0,d5

                   *if next char is not valid hex digit ret
02DC'              get_long_3
02DC'  610C            bsr.s get_hex_no
02DE'  6608            bne.s get_long_2

                   *else shift previous result left 4,add new no
                   *and loop
02E0'  E985            asl.l #4,d5
02E2'  DA84            add.b d4,d5
02E4'  60F6            bra.s get_long_3
02E6'              get_long_2
02E6'  7000            moveq #0,d0
02E8'  4E75            rts

                   *sub routine to convert ascii char to hex
                   *test for char in range 0 to 9
02EA'              get_hex_no
02EA'  1818            move.b (a0)+,d4
02EC'  983C 0030       sub.b #'0',d4
02F0'  6D32            blt.s get_hex_no_1
02F2'  B83C 0009       cmp.b #9,d4
02F6'  6204            bhi.s get_hex_no_2
02F8'  7000            moveq #0,d0
02FA'  4E75            rts

                   *test for char in range A to F
02FC'              get_hex_no_2
02FC'  983C 0007       sub #7,d4
0300'  B83C 000A       cmp.b #10,d4
0304'  6D1E            blt.s get_hex_no_1
0306'  B83C 000F       cmp.b #15,d4
030A'  6204            bhi.s get_hex_no_3
030C'  7000            moveq #0,d0
030E'  4E75            rts

                   *test for char in range a to f
0310'              get_hex_no_3
0310'  983C 0020       sub #$20,d4
0314'  B83C 000A       cmp.b #10,d4
0318'  6D0A            blt.s get_hex_no_1
031A'  B83C 000F       cmp.b #15,d4
031E'  6204            bhi.s get_hex_no_1
0320'  7000            moveq #0,d0
0322'  4E75            rts

                   *if non hex char found reposition pointer
                   *and report unsuccessful conversion
0324'              get_hex_no_1
0324'  5388            subq.l #1,a0
0326'  70FF            moveq #-1,d0
0328'  4E75            rts

       =0040       file_buf_len equ 64
032A'  =0040       file_buf  ds.b file_buf_len
036A'  =0004       file_id   ds.l 1
036E'  =0004       file_base ds.l 1

                   *routine to load exec file into mem
                   *exit if first non space char is eol
0372'              do_load
0372'  6100 FF34       bsr blanks
0376'  6600 FDCC       bne debug2

                   *else get length of file name and open it
037A'  6100 0092       bsr get_length
037E'  72FF            moveq #this_job,d1
0380'  7600            moveq #0,d3
                       QDOS io_open
0382'  7001                    moveq #io_open&$FF,d0
0384'  4E42                    trap #io_open/$100
0386'  4A80                    tst.l d0
0388'  6712            beq.s do_load_1

                   *branch if opened ok else report error and exi
                    t
038A'              do_load_2
038A'  43FA FC94       lea channel,a0
038E'  2050            move.l (a0),a0
                       VECTOR ut_err,a2
0390'  3478 00CC               move.w ut_err,a2
0394'  4E92                    jsr (a2)
0396'  4A80                    tst.l d0
0398'  6000 FDAA       bra debug2

                   *next read file header,exit on error
039C'              do_load_1
039C'  43FA FFCC       lea file_id,a1
03A0'  2280            move.l d0,(a1)
03A2'  343C 0040       move.w #file_buf_len,d2
03A6'  363C FFFF       move.w #-1,d3
03AA'  43FA FF7E       lea file_buf,a1
                       QDOS fs_headr
03AE'  7047                    moveq #fs_headr&$FF,d0
03B0'  4E43                    trap #fs_headr/$100
03B2'  4A80                    tst.l d0
03B4'  66D4            bne.s do_load_2

                   *find largest available space in trans proc ar
                    ea
                   *get file length + data space and report out
                   *of memory if not enough room for file
                       QDOS mt_free
03B6'  7006                    moveq #mt_free&$FF,d0
03B8'  4E41                    trap #mt_free/$100
03BA'  4A80                    tst.l d0
03BC'  43FA FF6C       lea file_buf,a1
03C0'  2411            move.l (a1),d2
03C2'  D4A9 0006       add.l 6(a1),d2
03C6'  B282            cmp.l d2,d1
03C8'  6404            bcc.s do_load_3
03CA'  70FD            moveq #err_om,d0
03CC'  60BC            bra.s do_load_2

                   *no of bytes needed in d1 so allocate space fo
                    r file
03CE'              do_load_3
03CE'  2202            move.l d2,d1
03D0'  74FF            moveq #this_job,d2
                       QDOS mt_alchp
03D2'  7018                    moveq #mt_alchp&$FF,d0
03D4'  4E41                    trap #mt_alchp/$100
03D6'  4A80                    tst.l d0
03D8'  43FA FF94       lea file_base,a1

                   *save allocated base address and also put it i
                    nto
                   *prog counter in reg blk
03DC'  2288            move.l a0,(a1)
03DE'  43FA 042C       lea pc,a1
03E2'  2288            move.l a0,(a1)

                   *compute end addr and put it into a7 reg blk a
                    s
                   *initial stack pointer value
03E4'  D1C0            add.l d0,a0
03E6'  2301            move.l d1,-(a1)

                   *and then load file
03E8'  2248            move.l a0,a1
03EA'  41FA FF7E       lea file_id,a0
03EE'  2050            move.l (a0),a0
03F0'  45FA FF38       lea file_buf,a2
03F4'  2412            move.l (a2),d2
03F6'  76FF            moveq #-1,d3
                       QDOS fs_load
03F8'  7048                    moveq #fs_load&$FF,d0
03FA'  4E43                    trap #fs_load/$100
03FC'  4A80                    tst.l d0

                   *if loading error exit
                   *else close file channel and report ok
                   *then back to supervisor loop
03FE'  668A            bne.s do_load_2
                       QDOS io_close
0400'  7002                    moveq #io_close&$FF,d0
0402'  4E42                    trap #io_close/$100
0404'  4A80                    tst.l d0
0406'  4EBA 001E       jsr ok
040A'  6000 FD38       bra debug2

                   *subroutine gets length of filename
                   *a0 and a1 set to point to start of name
                   *loop until a space or eol char is found
                   *incrementing a1 each time,then file name
                   *length = a1-a0
040E'              get_length
040E'  2248            move.l a0,a1
0410'              get_len_1
0410'  5249            addq #1,a1
0412'  0C11 0020       cmpi.b #' ',(a1)
0416'  6706            beq.s get_len_2
0418'  0C11 000A       cmpi.b #$A,(a1)
041C'  66F2            bne.s get_len_1

                   *now insert file name length in front of name
                    in
                   *standard QDOS string format
041E'              get_len_2
041E'  93C8            suba.l a0,a1
0420'  5548            subq #2,a0
0422'  3089            move.w a1,(a0)
0424'  4E75            rts

                   *subroutine prints ok to console
0426'              ok
0426'  41FA FBF8       lea channel,a0
042A'  2050            move.l (a0),a0
042C'  43FA 000C       lea msg_ok,a1
                       VECTOR ut_mtext,a2
0430'  3478 00D0               move.w ut_mtext,a2
0434'  4E92                    jsr (a2)
0436'  4A80                    tst.l d0
0438'  4E75            rts
043A'  0006        msg_ok    dc.w 6
043C'  2E2E 2E6F 6B0A          dc.b '...ok',$0A
0444'                  cnop 0,4

                   *single step routine
       =8000       trace     equ $8000 trace bit

                   *first save all regs
                   *stack a6 and load start addr of mem block
                   *where regs are to be saved in a6
0444'              single_step
0444'  2F0E            move.l a6,-(a7)
0446'  4DFA 0032       lea save,a6

                   *then save all regs except a6 & a7
044A'  48D6 3FFF       movem.l d0-d7/a0-a5,(a6)

                   *then save a6 and a7
044E'  41FA 0062       lea save+56,a0
0452'  20DF            move.l (a7)+,a6
0454'  20CE            move.l a6,(a0)+
0456'  208F            move.l a7,(a0)

                   *next enter supervisor mode
                   *and stack image of prog counter
```

(listing continued on page 118)

68008 QL

4 FREE WAFERS INCLUDING SPECTRAL WRITER WORD PROCESSOR

As a result of SMT's purchasing power, we are now able to offer the superb Rotronics Wafadrive data storage system for only £99.95*. But hurry! Stocks will not last long. You've read the reviews. Now's your best chance to buy!



SMT Special offer!

Wafadrive only £99-95*

*Plus VAT and £2.00 post and packing

Integrated System

The Wafadrive is a complete system which contains the micro interface, two 128K drives, RS232 and Centronics ports, all in one attractively-styled, compact unit. There is a minimum of connecting leads and no extra boxes to clutter the desk top. Like the majority of professional systems, the units are dual drive. This offers the optimum balance between system flexibility and cost. Built-in serial and parallel interfaces allow the direct connection of just about any popular printer.

Fast and Reliable

The Wafadrive achieves very fast loading and saving, but not at the expense of reliability. Extensive research and the use of high grade materials ensure that the Wafadrive will give years of dependable operation. Data integrity is on a par with floppy disk. Three fully interchangeable blank wafers in three sizes – 128K, 64K and 16K – are also included. Low capacity wafers give faster access. They are therefore most suitable for program development applications. The high capacity wafers are suitable for more general data storage. Loading rate is well over ten times as fast as cassette!

Software

Armed with the comprehensive user manuals, blank wafers and superb Spectral Writer word processor, you'll have a professional system to be proud of. See the coupon below for details of the rapidly growing range of software to enable the programmer and games player to exploit the Wafadrive system to the full.

Buy Now!

Cheap, reliable mass data storage for the Spectrum has finally arrived – make sure you benefit from SMT's special offer. Clip the coupon below and your order will receive immediate attention.

FOR USE WITH THE SINCLAIR SPECTRUM

Please send me (enter as appropriate):

____	ROTRONICS Spectrum Wafadrive(s)	@ £117.24 each	______
____	16K wafer(s)	@ £3.45 each	______
____	64K wafer(s)	@ £3.65 each	______
____	128K wafer(s)	@ £3.95 each	______
____	Heathrow/Nightflite II	@ £14.95 each	______
____	Space Wars/Seiddab Attack/Luna Attack	@ £14.95 each	______
____	Bear Bovver/Wong's Loopy Laundry/World Cup	@ £9.95 each	______
____	Starbike	@ £7.95 each	______
____	ZAP machine code	@ £19.95 each	______
____	Sherlock	@ £14.95 each	______
____	RS232 lead	@ £9.95 each	______
____	Centronics lead	@ £9.95 each	______
	All prices include VAT and P&P	TOTAL £	______

☐ I enclose a cheque/PO* made payable to SMT

☐ debit my Access/Barclay card* account no.

(*delete as applicable)

Signature: ______

Name: ______

Address: ______

Send to (no stamp required): **SMT, FREEPOST, Greens Norton, Towcester, Northants, NN12 8BR**

Please allow 28 days for delivery.

SMT

VISA

YC/4/85

Mail Order Protection Society registered number 3282/152

(listing continued from page 116)

68008 QL

```
0456'  43FA 0390         lea pc,a1
045A'  2F11              move.l (a1),-(a7)

                  *must set trace bit in memory image of status
reg
                  *and stack the status reg image
0460'  43FA 036C         lea status,a1
0464'  0051 8000         ori #trace,(a1)
0468'  3F11              move.w (a1),-(a7)

                  *now back to user mode
046A'  0A7C 2000         eori #super,sr

                  *and load all regs with their memory image
046E'  41FA 036C         lea d_reg_blk,a0
0472'  4CD8 FFFF         movem.l (a0),d0-d7/a0-a7

                  *back to supervisor mode
0476'  4E40              trap #0

                  *executing a rat from exception will cause the
 status
                  *reg and program counter to be pulled from the
                  *stack and the instruction pointed to by the p
c to be
                  *executed.Then as the trace bit is set an exce
ption will
                  *be forced,causing a jump to trace_ret,thus gi
ving a
                  *single step facility
0478'  4E73              rte

047A'  =0040      save       ds.l 16     regs saved here

                  *upon completion of a single step the exceptio
n
                  *will force a jump to here
                  *so first enter user mode & pull status reg o
f the stack
       =2000      super      equ #2000

04BA'             trace_ret
04BA'  0A7C 2000         eori #super,sr

                  *then save all regs in memory
                  *stack a6,save d0/a5 in reg blk
04BE'  2F0E              move.l a6,-(a7)
04C0'  4DFA 030A         lea d_reg_blk,a6
04C4'  48D6 3FFF         movem.l d0-d7/a0-a5,(a6)

                  *restore a6 and save it and a7
04C8'  2C5F              move.l (a7)+,a6
04CA'  41FA 0338         lea d_reg_blk+56,a0
04CE'  20CE              move.l a6,(a0)+
04D0'  208F              move.l a7,(a0)

                  *now back to supervisor mode
04D2'  4E40              trap #0

                  *and pull status reg off stack
04D4'  43FA 0336         lea status,a1
04D8'  329F              move.w (a7)+,(a1)

                  *next pull the pc counter of the stack
04DA'  43FA 0330         lea pc,a1
04DE'  229F              move.l (a7)+,(a1)

                  *and now back to user mode again
04E0'  0A7C 2000         eori #super,sr

                  *finally restore all regs and  back to supervi
sor
04E4'  4CFA FFFF FF92    movem.l save,d0-d7/a0-a7
04EA'  6000 FC50         bra debug2

                  **flags routine
                  *point to condition codes in a1
                  *get next char of line
04EE'             flags
04EE'  43FA 0321         lea status+1,a1
04F2'  1018              move.b (a0)+,d0

                  *if X or x toggle X flag  then back to supervi
sor
04F4'  0C00 0078         cmp.b #'x',d0
04F8'  6706              beq.s flags_x1
04FA'  0C00 0058         cmp.b #'X',d0
04FE'  6608              bne.s flags_n
0500'             flags_x1
0500'  0851 0004         bchg #4,(a1)
0504'  6000 FC36         bra debug2

                  *similarly for the other flags
0508'             flags_n
0508'  0C00 006E         cmp.b #'n',d0
050C'  6706              beq.s flags_n1
050E'  0C00 004E         cmp.b #'N',d0
0512'  6608              bne.s flags_z
0514'             flags_n1
0514'  0851 0003         bchg #3,(a1)
0518'  6000 FC2A         bra debug2
051C'             flags_z
051C'  0C00 007A         cmp.b #'z',d0
0520'  6706              beq.s flags_z1
0522'  0C00 005A         cmp.b #'Z',d0
0526'  6608              bne.s flags_v
0528'             flags_z1
0528'  0851 0002         bchg #2,(a1)
052C'  6000 FC1A         bra debug2
0530'             flags_v
0530'  0C00 0076         cmp.b #'v',d0
0534'  6706              beq.s flags_v1
0536'  0C00 0056         cmp.b #'V',d0
053A'  6608              bne.s flags_c
053C'             flags_v1
053C'  0851 0001         bchg #1,(a1)
0540'  6000 FC02         bra debug2
0544'             flags_c
0544'  0C00 0063         cmp.b #'c',d0

0548'  6708              beq.s flags_c1
054A'  0C00 0043         cmp.b #'C',d0

                  *back to super if no valid flag found
054E'  6600 FBF4         bne debug2
0552'             flags_c1
0552'  0851 0000         bchg #0,(a1)
0556'  6000 FBEC         bra debug2

                  *memory change/display routine
       =0001      byte       equ 1
       =0002      word       equ 2
       =0004      long       equ 4

055A'  =0000      buf_mem    ds.b 0
055A'                 cnop 0,2
055A'  =0004      buf_stk    ds.l 1

                  *get .size.if not found assume .w
055E'             do_mem
055E'  7E02              moveq #word,d7
0560'  0C10 002E         cmpi.b #'.',(a0)
0564'  6620              bne.s do_mem_1
0566'  5248              addq #1,a0
0568'  0C10 0062         cmpi.b #'b',(a0)
056C'  6716              beq.s do_mem_2
056E'  0C10 0042         cmpi.b #'B',(a0)
0572'  6710              beq.s do_mem_2
0574'  0C10 006C         cmpi.b #'l',(a0)
0578'  6706              beq.s do_mem_3
057A'  0C10 004C         cmpi.b #'L',(a0)
057E'  6606              bne.s do_mem_1
0580'             do_mem_3
0580'  7E04              moveq #long,d7
0582'  6002              bra.s do_mem_1
0584'             do_mem_2
0584'  7E01              moveq #byte,d7

                  *get next char and exit if eol
0586'             do_mem_1
0586'  5248              addq #1,a0
0588'  6100 FD16         bsr blanks
058C'  6600 FBAE         bne debug2

                  *else get hex address in d5
0590'  6100 FD36         bsr get_long

                  *save it in a4(1st addr) and a5 (2nd addr)
0594'  2845              move.l d5,a4
0596'  2A45              move.l d5,a5

                  *get next char,if eol display single char
                  *if equals then branch to change
                  *if comma then get second addr
                  *else display single location
0598'  6100 FD0E         bsr blanks
059C'  6A72              bmi.s do_mem_4
059E'  1018              move.b (a0)+,d0
05A0'  0C00 003D         cmpi.b #'=',d0
05A4'  6700 00D6         beq do_mem_7
05A8'  0C00 002C         cmpi.b #',',d0
05AC'  6612              bne.s do_mem_4

                  *get 2nd addr,if not found display single loca
tion
05AE'  6100 FCF8         bsr blanks
05B2'  6A0C              bmi.s do_mem_4
05B4'  6100 FD14         bsr get_long
05B8'  2A45              move.l d5,a5

                  *if 1st addr > 2nd addr then swap
05BA'  BBCC              cmpa.l a4,a5
05BC'  6402              bcc.s do_mem_4
05BE'  C94D              exg a4,a5

                  *start of line,set up 16 bytes per line
05C0'             do_mem_4
05C0'  9DCE              suba.l a6,a6
05C2'  7C10              moveq #16,d6
05C4'  2A07              move.l d7,d5

                  *convert address to ascii
05C6'  41FA FF92         lea buf_mem,a0
05CA'  43FA FF8E         lea buf_stk,a1
05CE'  228C              move.l a4,(a1)
                         VECTOR cn_itohl,a2
05D0'  3478 00F2         +      move.w cn_itohl,a2
05D4'  4E92              +      jsr (a2)
05D6'  4A80              +      tst.l d0

                  *and print it
05D8'  41FA FA3C         lea channel,a0
05DC'  2050              move.l (a0),a0
05DE'  43FA FF72         lea buf_mem,a1
05E2'  343C 0008         move.w #8,d2
05E6'  363C FFFF         move #-1,d3
                         QDOS io_sstrg
05EA'  7007              +      moveq #io_sstrg&$FF,d0
05EC'  4E43              +      trap #io_sstrg/$100
05EE'  4A80              +      tst.l d0
05F0'  6100 0074         bsr print_space
05F4'  6100 0064         bsr print_equals
05F8'  6100 006C         bsr print_space

                  *now convert byte pointed to by a4
05FC'             do_mem_6
05FC'  41FA FF5C         lea buf_mem,a0
0600'  43FA FF58         lea buf_stk,a1
0604'  12DC              move.b (a4)+,(a1)
                         VECTOR cn_itohb,a2
0606'  3478 00FA         +      move.w cn_itohb,a2
060A'  4E92              +      jsr (a2)
060C'  4A80              +      tst.l d0

                  *and print it
060E'  41FA FA00         lea channel,a0
0612'  2050              move.l (a0),a0
0614'  43FA FF3C         lea buf_mem,a1
0618'  343C 0002         move.w #2,d2
061C'  363C FFFF         move.w #-1,d3
                         QDOS io_sstrg
0620'  7007              +      moveq #io_sstrg&$FF,d0
0622'  4E43              +      trap #io_sstrg/$100
0624'  4A80              +      tst.l d0

                  *loop if no of bytes for size not done
0626'  5345              subq #1,d5
0628'  6704              beq.s do_mem_5
062A'  534E              subq #1,d6
062C'  60CE              bra.s do_mem_6

                  *else print space as seperator
062E'             do_mem_5
062E'  6136              bsr.s print_space
0630'  2A07              move.l d7,d5

                  *exit if all printed
0632'  B9CD              cmpa.l a5,a4
0634'  6E1E              bhi.s do_mem_exit

                  *else test to see if eol needed
0636'  534E              subq #1,d6
0638'  66C2              bne.s do_mem_6
063A'  6124              bsr.s print_eol

                  *if eol then test for any key to break
063C'  363C 0001         move.w #1,d3
0640'  41FA F9D6         lea channel,a0
0644'  2050              move.l (a0),a0
                         QDOS io_fbyte
0646'  7001              +      moveq #io_fbyte&$FF,d0
0648'  4E43              +      trap #io_fbyte/$100
064A'  4A80              +      tst.l d0
064C'  B07C FFFF         cmp.w #err_nc,d0
0650'  6700 FF6E         beq do_mem_4

0654'             do_mem_exit
0654'  6108              bsr.s print_eol
0656'  6000 FAE4         bra debug2

                  *single char print sub routines
065A'             print_equals
065A'  123C 003D         move.b #'=',d1
065E'  600E              bra.s print_1
0660'             print_eol
0660'  123C 000A         move.b #$0A,d1
0664'  6008              bra.s print_1
0666'             print_space
0666'  123C 0020         move.b #' ',d1
066A'             print_1
066A'  363C FFFF         move.w #-1,d3
066E'  41FA F9A8         lea channel,a0
0672'  2050              move.l (a0),a0
                         QDOS io_sbyte
0674'  7005              +      moveq #io_sbyte&$FF,d0
0676'  4E43              +      trap #io_sbyte/$100
0678'  4A80              +      tst.l d0
067A'  4E75              rts

                  *get new contents of mem location in d5
067C'             do_mem_7
067C'  6100 FC4C         bsr get_long

                  *and move to mem location byte by byte
                   according to size
0680'  BE3C 0001         cmp.b #byte,d7
0684'  671A              beq.s do_mem_8
0686'  BE3C 0002         cmp.b #word,d7
068A'  670A              beq.s do_mem_9
068C'  E198              rol.l #8,d5
068E'  18C5              move.b d5,(a4)+
0690'  E198              rol.l #8,d5
0692'  18C5              move.b d5,(a4)+
0694'  4845              swap d5
0696'             do_mem_9
0696'  E058              ror.w #8,d5
0698'  18C5              move.b d5,(a4)+
069A'  E158              rol.w #8,d5
069C'             do_mem_8
069C'  1885              move.b d5,(a4)
069E'  6000 FA9C         bra debug2

                  *print all regs routine
                  *print data reg now
06A2'             print_reg
06A2'  43FA 0166         lea data_msg,a1
06A6'  41FA F970         lea channel,a0
06AA'  2050              move.l (a0),a0
                         VECTOR ut_mtext,a2
06AC'  3478 00D0         +      move.w ut_mtext,a2
06B0'  4E92              +      jsr (a2)
06B2'  4A80              +      tst.l d0
06B4'  6600 0200         bne error

                  *print data regs from reg blk
06B8'  43FA 010A         lea d_reg_blk,a1
06BC'  6170              bsr.s p_regs
06BE'  6600 01F2         bne error

                  *print addr regs now
06C2'  43FA 010C         lea addr_msg,a1
06C6'  41FA F950         lea channel,a0
06CA'  2050              move.l (a0),a0
                         VECTOR ut_mtext,a2
06CC'  3478 00D0         +      move.w ut_mtext,a2
06D0'  4E92              +      jsr (a2)
06D2'  4A80              +      tst.l d0
06D4'  6600 01DC         bne error

                  *print addr regs from reg blk
06D8'  43FA 0104         lea a_reg_blk,a1
06DC'  6150              bsr.s p_regs
06DE'  6600 01D2         bne error

                  *print pc and flags msg
06E2'  43FA 0186         lea others_msg,a1
06E6'  41FA F930         lea channel,a0
06EA'  2050              move.l (a0),a0
                         VECTOR ut_mtext,a2
06EC'  3478 00D0         +      move.w ut_mtext,a2
06F0'  4E92              +      jsr (a2)
06F2'  4A80              +      tst.l d0

                  *convert pc contents to hex ascii
                  *and flags to binary ascii
06F4'  41FA 00E2         lea buf_others_pc,a0
06F8'  43FA 0106         lea pc,a1
06FC'  9DCE              suba.l a6,a6
                         VECTOR cn_itohl,a2
06FE'  3478 00F2         +      move.w cn_itohl,a2
0702'  4E92              +      jsr (a2)
0704'  4A80              +      tst.l d0
0706'  41FA 00E0         lea buf_others_flags,a0

0712'  43FA 00FC         lea status,a1
0716'  9DCE              suba.l a6,a6
                         VECTOR cn_itobw,a2
0718'  3478 00F6         +      move.w cn_itobw,a2
071C'  4E92              +      jsr (a2)
071E'  4A80              +      tst.l d0
0720'  41FA F8F6         lea channel,a0
0724'  2050              move.l (a0),a0
0726'  43FA 00B0         lea buf_others_start,a1

                  *print pc and flags then back to
                   supervisor
                         VECTOR ut_mtext,a2
072A'  3478 00D0         +      move.w ut_mtext,a2
072E'  4E92              +      jsr (a2)
0730'  4A80              +      tst.l d0
0732'  6000 FA10         bra debug2

                  *print regs subroutine
0736'             p_regs
0736'  7E07              moveq #7,d7
0738'  41FA 00A2         lea buf_temp,a0
073C'  9DCE              suba.l a6,a6
073E'             p_loop
                         VECTOR cn_itohl,a2
073E'  3478 00F2         +      move.w cn_itohl,a2
0742'  4E92              +      jsr (a2)
0744'  4A80              +      tst.l d0
0746'  10FC 0020         move.b #' ',(a0)+
074A'  51CF FFF2         dbra d7,p_loop
074E'  41FA F8C8         lea channel,a0
0752'  2050              move.l (a0),a0
0754'  43FA 008C         lea buf_start,a1
                         VECTOR ut_mtext,a2
0758'  3478 00D0         +      move.w ut_mtext,a2
075C'  4E92              +      jsr (a2)
075E'  4A80              +      tst.l d0
0760'  4E75              rts

0762'  0049       buf_start  dc.w 73
0764'  =0048      buf_temp   ds.b 72
07AC'  0A                 dc.b $0A
07AE'                     cnop 0,2
07AE'             buf_others_start
07AE'  001A               dc.w 26
07B0'             buf_others_pc
07B0'  =0008              ds.b 8
07B8'  20                 dc.b ' '
07B9'             buf_others_flags
07B9'  =0010              ds.b 16
07C9'  0A                 dc.b $0A
07CA'                     cnop 0,4
07CC'  =0020      d_reg_blk  ds.l 8
07EC'  =0020      a_reg_blk  ds.l 8
080C'  =0004      pc         ds.l 1
0810'  =0002      status     ds.w 1
0812'  0045       data_msg   dc.w 69
0814'  2020 2044 3020 2020    dc.b '  D0
       20
0819'  2020 2044 3120 2020    dc.b '  D1
       20
081E'  2020 2044 3220 2020    dc.b '  D2
       20
0823'  2020 2044 3320 2020    dc.b '  D3
       20
0828'  2020 2044 3420 2020    dc.b '  D4
       20
082D'  2020 2044 3520 2020    dc.b '  D5
       20
0832'  2020 2044 3620 2020    dc.b '  D6
       20
0837'  2020 2044 37           dc.b '  D7'
083C'  0A                 dc.b $0A
083E'                     cnop 0,2
083E'  0045       addr_msg   dc.w 69
0840'  2020 2041 3020 2020    dc.b '  A0
       20
0845'  2020 2041 3120 2020    dc.b '  A1
       20
084A'  2020 2041 3220 2020    dc.b '  A2
       20
084F'  2020 2041 3320 2020    dc.b '  A3
       20
0854'  2020 2041 3420 2020    dc.b '  A4
       20
0859'  2020 2041 3520 2020    dc.b '  A5
       20
085E'  2020 2041 3620 2020    dc.b '  A6
       20
0863'  2020 2041 37           dc.b '  A7'
0868'  0A                 dc.b $0A
086A'                     cnop 0,2
086A'             others_msg
086A'  001A               dc.w 26
086C'  2020 2050 4320 2020    dc.b '    PC      X,N,,Z,V,,,C,PROFG'
       2020 2020 2020 4F4E
       4F26 2E2E 5846 5A56
       4380                   ,$0A
0886'                     cnop 0,2

                  *quit routine
                  *push zero error code and ret to
                   basic
0886'             quit
0886'  2F3C 0000 0000    move.l #0,-(a7)
088C'  6002              bra.s exit

                  *error exit
                  *leave error code,close console and
                   remove job
088E'             error
088E'  2F00              move.l d0,-(a7)
0890'             exit
0890'  41FA F786         lea channel,a0
                         QDOS io_close
0894'  7002              +      moveq #io_close&$FF,d0
0896'  4E42              +      trap #io_close/$100
0898'  4A80              +      tst.l d0
089A'  72FF              moveq #this_job,d1
089C'  261F              move.l (a7)+,d3
                         QDOS mt_frjob
089E'  7005              +      moveq #mt_frjob&$FF,d0
08A0'  4E41              +      trap #mt_frjob/$100
08A2'  4A80              +      tst.l d0
08A4'  7000              moveq #0,d0
08A6'  4E75              rts
                         end

No errors found in this Assembly
```

Write programs to unlock the multitasking power of your QL!

*Use Metacomco's much acclaimed **Development Kits** to create the software for your QL: Use **Assembler** for its speed, and for access to the QL's many features; use **BCPL** for systems programming; writing games, utilities, and applications packages; use **LISP** for manipulating data structures, and for exploring the world of artificial intelligence.*

RELEASE THE POWER OF THE QL

ASSEMBLER £39.95 INC. VAT + P&P

> Standard Motorola 68000 mnemonics.
> English error messages.
> Produces code which can be EXECed, and run as a concurrent job.
> External references allow linkage to high level languages.
> Integral linker allows assembler programs to be linked together.
> Macro expansions.
> Position independent, absolute or relocatable code can be produced.
> Conditional assembly.
> Large range of directives.
> Fully formatted listings.
> 30 character variable names and 32-bit values.

ENHANCED! NOW INCLUDES ASSEMBLER LINKER* AND NEW LOW PRICE **£39.95**

BCPL £59.95 INC. VAT + P&P

> True compiler.
> Multiple BCPL programs can run concurrently.
> Full runtime library includes interfaces to QDOS graphics, file operations, window handling, etc.
> Exception handling provides diagnostics for debugging.
> Link loader allows linkage of separately compiled segments.
> Assembler and BCPL modules can be linked together.
> 32-bit variables use the full QL address space.

Each language is supplied as a complete Development Kit which includes the software (on a Microdrive Cartridge), Metacomco's popular full screen editor, and user manual.

LISP £59.95 INC. VAT + P&P

> Full support of QL features including window graphics, and screen handling.
> Compatible with Acornsoft Lisp for the BBC micro.
> Interpreter.
> Turtle graphics.
> Structure editor allows alteration of data structures.
> Prettyprinter displays programs in structured format.
> Tracer to aid in debugging.
> Garbage collector automatically recovers spare memory space.
> 28-bit integers and 250-character names.

The Metacomco Development Kits are available from HMV, Menzies and many other leading retailers, or direct from Metacomco.

METACOMCO

26 Portland Square, Bristol BS2 8RZ.
Tel: Bristol (0272) 428781

*Existing registered users can obtain an upgrade. Contact Metacomco for details.

QL & QDOS are the trade names of Sinclair Research Ltd.

NAME ____________

ADDRESS ____________

____

POSTCODE ____________

TEL. NO. ____________

SIGNATURE ____________

Phone today, or post this coupon to: Metacomco, 26 Portland Square, Bristol BS2 8RZ.

Please send me:

The Assembler Development Kit at £39.95 ____

The BCPL Development Kit at £59.95 ____

The LISP Development Kit at £59.95 ____

I enclose a cheque for £ ____ or debit my ACCESS/VISA Account No.

Card expiry date: ____

Price includes VAT, postage and packing UK mainland only. Delivery allow up to 28 days.

YC3

Listing 2.

```
41767 43073E072F072B072607 = 292
41777 0C070707070107FD06EC06 = 540
41787 CB06BF06B306AE06A006 = 937
41797 8F067C06590647064506 = 528
41807 3D063806350629062306 = 276
41817 20067F05710568056505 = 503
41827 5F055805540533052605 = 381
41837 1B05F904F604DE04D904 = 982
41847 D604CF04CB04B904B204 = 1007
41857 AD04A804A5049D049804 = 835
41867 950490048B047F047804 = 699
41877 75045A04452044A04400 4 = 447
41887 3A0420041B0412040B04 = 166
41897 0304FC03E903DC03CD03 = 929
41907 8D017C0170016B016701 = 592
41917 62015F01520143013C01 = 407
41927 330129012501200118 01 = 193
41937 180115011201 0B010801 = 87
41947 05010201FD00FA00F700 = 759
41957 F400ED00D800D400BC00 = 1097
41967 8A006C0064005B003100 = 486
41977 2C002900240021000000 = 154
41987 2124FF194E2346237881 = 864
41997 2814E56B62094E2346E5 = 915
42007 6B62094D44E17028 71E1 = 1077
42017 18E4012900218103C3D1 = 911
42027 BCA401C345000060E1196 = 804
42037 01CD788B252004050513 = 615
42047 131ACD5ABB1310F9C9DD = 1233
42057 E5E1FE0230E20E002807 = 1053
42067 FE0320DA4E23233EC0A1 = 1070
42077 32BF03CBB95E2356ED53 = 1167
42087 B903235E2356ED53B503 = 942
42097 CB412017CD11BCD83E07 = 1018
42107 CDB4BB47C52100001118 = 914
42117 27CD66BBC11813217007 = 921
42127 237EFEFF280A064DCD2B = 1049
42137 BD38F310F9C9C5CD06B7 = 1547
42147 57CB49CC09B9CD12B95F = 1264
42157 CB51280ACB590E012801 = 682
42167 0DCD0FB9C1D5C5CDDE00 = 1448
42177 C1CB41200CCD78BB7085 = 1366
42187 CDB4BBE1CD75BBC1C318 = 1718
42197 B92AB503ED4BB703A7ED = 1313
42207 42C90640CB4120070618 = 674
42217 C5CD6CBBC1C571BF037E = 1440
42227 E6CC772AB50322B70321 = 1032
42237 5E0722BB030EFECD1104 = 819
42247 21470722BB03CDD30028 = 791
42257 2309ED43B50322B703CD = 957
42267 EA042ABB0323232 2BB03 = 766
42277 E5AFCDBE06D1CDD30028 = 1470
42287 05EB232318EB2AB9030B = 810
42297 ED42C130053ABF03E60C = 1043
42307 282DCD6F010428A30510 = 630
42317 A0CB41280F3E07CD9001 = 902
42327 CD18BBCBAFFE432881C9 = 1485
42337 21500722BB031E75CD89 = 833
42347 060DCD6F0118E5214707 = 700
42357 16287E3620CB412805CD = 792
42367 900118 03CD5ABB231520 = 742
42377 EDCB41C87E23CD90017E = 1342
42387 CD2BBD30FBC90D0A5379 = 1164
42397 6E746178206572726F72 = 1029
42407 44495341535345D424C = 743
42417 45D20080C2C3C4C5C8CC = 1593
42427 28484CA9C1C148CCC142 = 1278
42437 C344C58A53D041C62842 = 1258
42447 43A92844445A92804A928 = 835
42457 04A94EDADA4EC3C350CF = 1442
42467 50C5D0CD41444420A041 = 1148
42477 444320A0535542 20A053 = 836
42487 424320A0414E4420A058 = 816
42497 4F5220A04F522020A043 = 805
42507 502020A0524554 20A045 = 800
42517 58D84A502020202080AA9 = 773
42527 4C44202020535 02C8A4A = 659
42537 502020208448414CD44F = 812
42547 5554202028022 92CC149 = 626
42557 4E20202041 2C2802A945 = 563
42567 5820202028535029 2C8A = 610
42577 4558202020444 52C48CC = 710
42587 44C945C94E4FD0455820 = 1093
42597 2020414 62C4146A2444A = 682
42607 4E5A20834A5220202083 = 714
42617 4A522020A04A502020A0 = 758
42627 43414C4CA04C442020A0 = 812
42637 494E4320A0444543 20A0 = 806
42647 504F5020A0505553 48A0 = 911
42657 494E2020A04F555420A0 = 815
42667 524C4320A0525243 20A0 = 840
42677 524C2020A052522020A0 = 770
42687 534C4120A0535241 20A0 = 838
42697 494E56414C49444 2034F = 697
42707 44C5535 24C20A02849 58 = 879
42717 2B02A92849592B02A952 = 712
42727 535420A049D849D94CC4 = 1210
42737 43D049CE4FD4C9C449D2 = 1525
42747 44D2B0B1B2B3B4B5B6B7 = 1714
42757 4249 5420A0524553 20A0 = 841
42767 53455420A0494D2020A0 = 802
42777 5252C4524CC4CECYD228 = 1371
42787 43A94441 54C1524C43C1 = 1064
42797 525243C1524CC15252C1 = 1132
42807 4441C14350CC53 43C643 = 1072
42817 43C6434 14C4C20844E45 = 860
```

AMSTRAD DIS



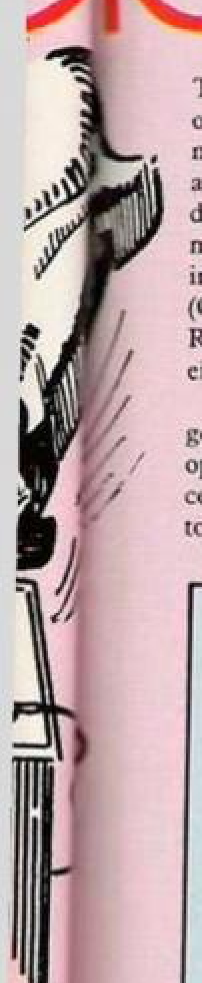
Kevin Probert with a more subtle way of disassembling than the picture shows.

Listing 1.

```
10  MEMORY 41766:LOAD "!CODE",41767:GOTO 110
20  SAVE "HEX LOADER":SAVE "CODE",b,41767,2136:END
100 MEMORY 41766
110 MODE 1
120 DIM check(2)
130 LOCATE 1,24:INPUT;"START ADDRESS ";address:CLS
140 LOCATE 1,24:PRINT address
150 INPUT ;"CODE,CHECKSUM ",a$,check(2)
160 check(1)=0
170 FOR n=1 TO LEN(a$) STEP 2
180 a=VAL("&"+MID$(a$,n,2))
190 check(1)=check(1)+a
200 POKE address+n-1,a
210 NEXT
220 IF check(1)=check(2) THEN 240
230 LOCATE 1,24:PRINT "CHECKSUM ERROR.PLEASE INPUT AGAIN":GOTO 260
240 address=address+10
250 LOCATE 10,24:PRINT UPPER$(a$);TAB(32)"= ";check(2)
260 LOCATE 1,25:PRINT SPACE$(41)
270 GOTO 140
```

```
42827 C74C524F4D204FCE4C52 - 980
42837 4F4D204F46C655524F4D - 858
42847 204F4EA055524F4D204F - 783
42857 4646A050524553532022 - 763
42867 432220544F20434F4E54 - 636
42877 494E55C5524F4D20A052 - 945
42887 4F4D2020AB000200020E - 409
42897 0E7002404040404040BF - 703
42907 40404040404040404000 - 576
42917 0800002A020002FFAFCF - 691
42927 8FCF8F0C0F0000000000 - 520
42937 00000000000000002E80 - 174
42947 78E6F8180378E6380F0F - 1061
42957 0FC9CDC0035F16001978 - 878
42967 E607070707C64632DF03 - 802
42977 CB00C920161AFEFC380A - 1056
42987 CD970411736E0F0F183D - 717
42997 CBDE11736F18401E71C3 - 1094
43007 89061E660C0CCC890679 - 767
43017 FE07C8ED5BB90337ED52 - 1351
43027 C821BF03CB562834CB96 - 1161
43037 3ABD0347ED5BB703FEDF - 1312
43047 30BB1BFED728111A1173 - 946
43057 70CB7F28011CCB772801 - 874
43067 14C34306EB1E77CD8906 - 1020
43077 7EE6C007076F37C3DE06 - 1151
43087 CB5ECB9E7ECD97044328 - 1251
43097 081E76CD89066018E9CB - 1060
43107 77209BCB6F2839CB4728 - 1031
43117 084678E607FE0620392B - 827
43127 22B703CDC003FE06282E - 966
43137 CDC10320050D1E3F1807 - 575
43147 1E5BCD67041E54CD5806 - 848
43157 1E01CD64062AB7035E23 - 699
43167 22B703C9CB4728102189 - 921
43177 03CDCC032049CD97041E - 910
43187 45C38906CB67283DCDC0 - 1211
43197 03FE143812FE1830ECCB - 1116
43207 5020E8114C4CCD64060D - 837
43217 C34706CB5F28DA21A103 - 1025
43227 CDCC0328D211980570E6 - 1186
43237 0F6F2600EB195E19E55E - 866
43247 19E32235E23562366C911 - 857
43257 410521BF0378FEC0381D - 940
43267 FEED2814FECB280EFEFD - 1569
43277 2806FEDD20D0CBCECBC6 - 1571
43287 1804CBEECBE6C311042E - 1164
43297 06FE8030C12E1111DD05 - 935
43307 FE403809FE7420B41E2A - 1039
43317 C38706E60720A7CB6828 - 1121
43327 A32E1018A34042444740 - 753
43337 4BD409384A3C3F4040CC - 889
43347 010249CD89067832BD03 - 786
43357 D6C76F37CDDE0678D681 - 1475
43367 326B0521BE03CB00C823 - 826
43377 CBD6CDBA0678FED72806 - 1449
43387 3E0430023E06C3BE06C2 - 769
43397 2515BF3B8CBD361504BD - 905
43407 29B7371504B23C8CB4A5 - 1027
43417 B26E101316191C1D1F29 - 499
43427 080B1E21001500218C3D - 337
43437 0165903E65018C200E8C - 736
43447 8830138C907089256206 - 1013
43457 5FD47C1E0E8C70388C13 - 942
43467 04386308CB6828041E5C - 640
43477 1874CB60285C54266118 - 814
43487 5F121316191B1D1F220A - 310
43497 2225281315171A262959 - 368
43507 3153388C044F3891884A - 822
43517 398C473901443A014238 - 575
43527 01024267341D0E8C3838 - 519
43537 8891333A8C313511032D - 697
43547 380101041D0802CD5806 - 400
43557 CDC5035AFE04DC89060C - 1128
43567 5CCB70201E1D1830CD43 - 842
43577 06CDB806184DCD89065A - 940
43587 5418010DCD89065ACD58 - 853
43597 060C0D5CC07CFE012810 - 750
43607 FE053866CDC503CB7B28 - 1188
43617 08CBDB1F180378E40783 - 944
43627 5FFE07280EFE0A280AFE - 978
43637 0F2014CB782810180D3A - 541
43647 BF03CB472807CBF3CB4F - 1243
43657 20011CE5D5C50E6421B0 - 1023
43667 0123CB7E28FB1D20F823 - 1000
43677 7ECB7F2013CDA40618F5 - 1151
43687 CBBFFE0A28BE3010E5CD - 1384
43697 BE06E1C9CDA406C1D1E1 - 1624
43707 0CC03E2C187ECD9704FE - 1074
43717 03281D3816FE0653CD97 - 849
43727 04636A3819CBBCCBB428 - 1104
43737 13CBFCCBF4180D6B2600 - 1103
43747 18091600CB7B28011519 - 468
43757 A73ABF03CB7F28310E01 - 853
43767 30010D11F0D8CD110711 - 781
43777 18FCCD1107119CFFCD11 - 1155
43787 071EF6CD11077D180AAF - 846
43797 3C1938FCED523D28010C - 826
43807 5FAFB9C87B18113803CD - 1083
43817 2907457CCDC1031FCD34 - 962
43827 077CE60FC630FE3A3802 - 992
43837 C607E52ADB03772322BB - 1041
43847 03E1C920202020202020 - 653
43857 20202020202020202020 - 320
43867 20202020202020202020 - 320
43877 20202020202020202020 - 320
43887 2020200A0D00FF000000 - 374
43897 0000000000000000 - 0
```

DISASSEMBLER



THIS DISASSEMBLER occupies less than 2K once loaded and can be located anywhere in memory. It can be called from within Basic as an external command and (a) correctly disassembles all Z80 op codes using standard mnemonics including the RST instructions as implemented in the 464 Operating System (OS); (b) disassembles routines in Rom or Ram; (c) sends output in hex or decimal to either screen or printer.

Those who have tried to Peek the Rom to get a glimpse of the O.S. will appreciate option (b) since Peek always returns the contents of Ram. The program can be altered to a certain degree to suit user requirements.

The program has been implemented as an RSX — Resident System Extension — and once loaded sits above Himem. However, since Himem can vary dependent on space reserved for expansion Roms, for example, disc Rom or user machine-code routines then an RSX needs to be relocatable. This is achieved by an additional routine which is called before the program is "logged on" with the OS and makes use of the fact that a Call from within Basic enters the routine with the DE register pair containing the argument of the Call instruction. For those interested in using the RSX facility, the following illustrates how "logging on" is achieved:

LD BC,NNNN	Address of Command Table Pointer
LD HL,NNNN	Address of 4 bytes for OS to use as workspace
JP BCD1	*Log on external command(s) with OS*
COM TAB POINTER	2 bytes holding start address of keyword table
JP NNNN	FLASH routine
JP NNNN	INVERSE routine
etc	
FLAS H + 80h	Last letter in each keyword has bit 7 set
INVERS E + 80h	
etc	
0	End of keywords marker

Once logged on an external command server is recognised by preceding the keyword with shifted @

:FLASH

Parameters can also be passed to the external command server routine in the following fashion:

:FLASH,400,23

On entry to the routine the A register holds the number of parameters being passed, the index register IX points to an area in memory where the parameters are stored in two-byte integer form in the reverse order to that in which they were entered, that is, from the above example:

A = 2
IX + 0 = 23
IX + 1 = 0
IX + 2 = 144
IX + 3 = 1 (1*256 + 144 = 400)

Strings can also be passed to the external command server routine: in this case IX

(continued on page 123)

Table 1.

	SET (1)	RESET (0)
Bit 0 (1)	Output to printer	Output to screen
Bit 1 (2)	LROM On	LROM Off
Bit 2 (4)	UROM On	UROM Off
Bit 3 (8)	Disc ROM	BASIC ROM
Bit 4 (16)	Not used	
Bit 5 (32)	Not used	
Bit 6 (64)	Output as DATA	Output as code
Bit 7 (128)	Output in decimal	Output in hex

Bit 0. Self-explanatory.
Bit 1. Selects what will be disassembled (LRom or Ram) when addressing memory between 0000h and 3FFFh.
Bit 2. Selects what will be disassembled (URom or Ram) when addressing memory between C000h and FFFFh.
Bit 3. When Bit 2 is Reset, this Bit is ignored. When Bit 2 is Set, this Bit selects the URom which will be addressed.
Bit 6. When Reset memory between the start and finish addresses is disassembled; when Set this memory area is output as Data.
Bit 7. Self-explanatory.

The next move is up to you...



All American ADVENTURES

All American Adventures Ltd., Unit 10, Parkway Industrial Estate, Heneage Street, Birmingham B7 4LY. Telephone: 021-359 3020.

Listing 3.

```
300 MODE 1
310 LOCATE 1,25:INPUT;"START ADDRESS ";address:CLS
320 LOCATE 1,25
330 c$="":check=0
340 FOR n=0 TO 9
350 a=PEEK(address+n)
360 check=check+a:c$=c$+HEX$(a,2)
370 NEXT
380 PRINT address;TAB(10);c$;TAB(32);"= ";check
390 a$=INKEY$:IF a$="" THEN 390
400 address=address+10:GOTO 330
```

(continued from page 121)

points to an address which in turn points to a three-byte string describer where byte 0 holds the length of the string, bytes 1 and 2 hold the address where the string is stored.

I have selected a method of entering the program often seen in this magazine, because I have found this method to produce the least number of errors. The program should be entered with all expansion Roms disconnected. This can be checked by:

PRINT HIMEM

which should return the value 43903. Enter the hex loader program in listing 1 and when complete type Run 100. Input as prompted the code in listing 2, code and Checksum being separated by a comma. The code can be Saved at any stage of its development by Escaping from the loader program and typing Run 20. It can then be reLoaded at a later date by

RUN"

Once all code has been entered, delete the loader program and enter the Basic Load/Save program in listing 4. Save the completed program to tape by typing Run 4. Enter as a direct command

CALL 41987

and if the ready prompt appears go on to use the program as described later in this article. If the system crashes, reset the system then enter:

MEMORY 41766: LOAD "DISASSEMBLER":LOAD "!CODE", 41767

Once Loaded, check the code by entering listing 3 and Run 300. Compare the results to listing 2 and, after all errors have been corrected, reSave as above.

Once the program has been Saved in its final form it can be recalled by

RUN"

if no other program is in memory or by

CHAIN MERGE "DISASSEMBLER"

if a program in memory is to be retained. To use the disassembler the following syntax is required:

!disassembler, <start address>, <finish address>, <output state>

The start and finish addresses are mandatory, the output state optional defaulting to zero if omitted. All parameters can be either numeric or variable expressions.

For example, if it is required to send output to the printer, LRom on, URom off, as code in decimal, then the output state value is as follows:

$(1*1)+(1*2)+(0*4)+(0*64)+(1*128)=131$

When output to the screen is selected, the disassembler uses stream 7 and sets a window size on this stream 40 characters wide and 25 high. The disassembler must be used in Mode 1 or 2 or it will not operate.

As it stands, the disassembler produces mnemonics using upper case. For those who prefer mnemonics in lower case the program in listing 5 should be entered and run before saving the disassembler in its final form.

For those who intend to use a printer with the disassembler, various options are available. If using cut sheet paper then the number of lines per sheet can be set by Poking the value to 42210 — A4E2h. The disassembler will then wait at the end of the sheet for "C" to be pressed giving time to change the sheet.

If using continuous paper this value should be set to 255 — FFh. The area from 43893 (AB75h) to 43903 (AB7Fh) has been set aside for up to 10 control codes which the user can enter and are sent before printing starts. The control codes must be terminated by a marker byte 255 — FFh.

The only non-standard results produced by the disassembler are those of the RST 08h, 10h, 18h and 28h instructions. The 464 OS implements these op codes as "extensions" to the instruction set. When executed, the following two bytes are taken as an inline address; the RST 08h uses bits 14 and 15 to select the Rom state, the RST 10h uses these bits for Rom select.

The RST 18h instruction takes a third byte for Rom state/select but is thus able to jump to any location in memory in any Rom. Typical examples of the output produced by the disassembler for each of these RST instructions are given below plus their interpretation.

Mnemonic	Remark
RST 08,3B0F	JP 3B0F
UROM OFF LROM ON	Upper ROM disabled, Lower ROM enabled
RST 10,DFCC	CALL DFCC
ROM +2	Side CALL to an expansion ROM
RST 18,OD17	CALL OD17
UROM ON LROM OFF	Upper ROM enabled, Lower ROM disabled
ROM 199	Select ROM 199
RST 28,224C	JP 224C
LROM ON	Lower ROM enabled ■

Listing 4.

```
1 a=HIMEM-2137:MEMORY a:LOAD "!CODE",a+1:CLOSEIN
2 CALL a+221:REM RELOCATE AND LOG ON DISASSEMBLER
3 MEMORY a+241:DELETE 1-4:END
4 SAVE "DISASSEMBLER":SAVE "!CODE",b,41767,2136:END
```

Listing 5.

```
10 FOR address=42421 TO 42828
20 a=PEEK(address)
30 IF (a>64 AND a<91) OR (a>192
AND a<219) THEN 40 ELSE 50
40 a=a+32:POKE address,a
50 NEXT
```

Memory map.

Address	Routine
41767 (A327h)	Data for relocating routine.
41987 (A403h)	Relocating routine.
42019 (A423h)	Log on disassembler with OS.
42028 (A42Ch)	Data for log-on.
42033 (A431h)	Print syntax error.
42056 (A448h)	Entry to disassembler: check syntax.
42074 (A45Ah)	Set up output state, finish and start address.
42101 (A475h)	Set up stream and window size.
42124 (A48Ch)	Check if printer connected: send control codes.
42143 (A49Fh)	Set up required rom state.
42174 (A4BEh)	Call disassember start.
42177 (A4C1h)	Restore previous Rom state and stream: return.
42198 (A4D6h)	Check if start = current address.
42209 (A4E1h)	Disassembler start: set up No. of lines for screen/printer.
42236 (A4FCh)	Set up print position in print buffer.
42244 (A504h)	Call build up mnemonic in print buffer.
42253 (A50Dh)	Print address, op code and operand to print buffer.
42327 (A557h)	Wait for key press.
42354 (A572h)	Copy print buffer to current screen line/printer: return.
42393 (A599h)	Data for syntax error.
42407 (A5A7h)	Data for log-on assembler.
42420 (A5B4h)	Mnemonic keyword list/control codes.
42892 (A78Ch)	Data for valid IX/IY op codes.
42924 (A7ACh)	Data for valid ed op codes.
42932 (A7B4h)	Workspace for OS.
42936 (A7B8h)	Start, current and finish addresses.
42942 (A7BEh)	Print position in print buffer.
42944 (A7COh)	RST flags.
42946 (A7C2h)	Disassembler flag.
42947 (A7C3h)	Conversion routines.
42958 (A7CEh)	Check for valid IX/IY and ed op codes routine.
42980 (A7E4h)	RST 18/28 handler.
43009 (A801h)	Data handler.
43028 (A814h)	Entry to build up mnemonic in print buffer.
43035 (A81Bh)	RST 08/10 handler.
43114 (A86Ah)	CB op code handler.
43175 (A8A7h)	IX/IY op code handler.
43195 (A8BBh)	Ed op code handler.
43253 (A8F5h)	Check for CB,ED and IX/IY op codes.
43295 (A91Fh)	Op codes <64.
43332 (944h)	Data for op codes >= 192.
43419 (A99Bh)	Data for ed op codes.
43488 (A9EOh)	Data for op codes <64.
43575 (AA37h)	Various print routines.
43660 (AA8Ch)	Find mnemonic keyword and print to print buffer.
43713 (AAC1h)	Print decimal/hex number to print buffer.
43850 (AB4Ah)	Print buffer.
43890 (AB72h)	Carriage return: line feed.
43892 (AB74h)	Control codes.
43903 (AB7Fh)	End.

software that's hard to beat...

MSX
AMSTRAD
48K SPECTRUM

TASWORD TWO

The Word Processor ✱

"If you have been looking for a word processor, then look no further."
CRASH June 1984

"The number of on-screen prompts, together with the excellent manual, make it ideal – even for an absolute beginner."
PERSONAL COMPUTER WORLD September 1983

"Without doubt, the best utility I have reviewed for the Spectrum."
HOME COMPUTING WEEKLY April 1984

TASWORD TWO ZX 48K Spectrum **£13.90**

TASWORD MSX

The Word Processor

The Tasman Word Processor for MSX microcomputers.
All the features of the Spectrum version.

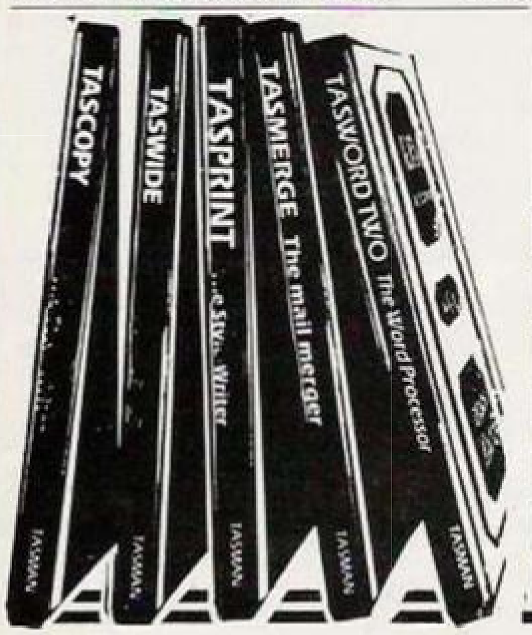
TASWORD MSX MSX Computers **£13.90**

TASCOPY

The Screen Copier

Screen copy software for Spectrum with Interface 1 and Amstrad CPC 464. Print high resolution screen copies (in a choice of sizes), and also large 'shaded' copies with different dot densities for the various screen colours. Tascopy supports all eight pin dot matrix printers with Epson type control codes, e.g. Epson RX-80 and FX-80, Shinwa CP-80, Mannesmann Tally MT-80, Star DMP 510/515, Brother HR5, and also Amstrad DMP 1 with the CPC 464.

TASCOPY ZX Spectrum £9.90
TASCOPY 464 Amstrad CP 464 £9.90



TASWORD 464

The Word Processor

The Amstrad implementation of Tasword Two plus many extra features.

TASWORD 464 Amstrad CPC 464 **£19.95**

TASMERGE

The Mail Merger

Transfer data from MASTERFILE to TASWORD TWO! Letters and forms typed on TASWORD TWO can be printed with addresses and data taken from MASTERFILE. The mail merge facility allows, for example, multiple copies of a letter to be printed, each containing a different name and address taken from your MASTERFILE data. To use TASMERGE you must have one or more microdrives as well as TASWORD TWO and MASTERFILE by Campbell Systems. (version 9 or later).

TASMERGE ZX 48K Spectrum **£10.90**

TASPRINT

The Style Writer

A must for dot-matrix printer owners! Print your program output and listings in a choice of five impressive print styles. TASPRINT utilises the graphics capabilities of dot-matrix printers to form, with a double pass of the printhead, output in a range of five fonts varying from the futuristic DATARUN to the hand-writing style of PALACE SCRIPT. TASPRINT drives all dot-matrix printers with bit image graphics capabilities and can be used to print TASWORD text files. TASPRINT gives your output originality and style!

TASPRINT ZX 48K Spectrum **£9.90**
TASPRINT Amstrad CPC 464 **£9.90**

TASWIDE

The Screen Stretcher

With this machine code utility you can write your own Basic programs that will, with normal PRINT statements, print onto the screen in the compact lettering used by TASWORD TWO. With TASWIDE you can double the information shown on the screen!

TASWIDE ZX 48K Spectrum **£5.50**

TASMAN PRINTER INTERFACE

Plug into your Spectrum and drive any printer fitted with the Centronics standard parallel interface. Supplied complete with ribbon cable and driving software. The user changeable interface software makes it easy to send control codes to your printer using the method so successfully pioneered with TASWORD TWO. The cassette also contains fast machine code high resolution full width SCREEN COPY SOFTWARE for Epson, Mannesmann Tally, Seikosha, Shinwa, Star, and Tandy Colour Graphic (in colour!) printers. Compatible with microdrives and ZX Interface 1.

PRICE **£39.90**

TASMAN SOFTWARE
All prices include VAT and post and packaging.
Telephone orders: Leeds (0532) 438301

✱ Available from larger branches of Boots.

Tasman SOFTWARE

Springfield House, Hyde Terrace, Leeds LS2 9LN. Tel: (0532) 438301

If you do not want to cut this magazine just write your order and post to:
TASMAN SOFTWARE, dept. YC, Springfield House, Hyde Terrace, Leeds LS2 9LN.
I enclose a cheque/P.O. made payable to Tasman Software Ltd. OR charge my ACCESS number

NAME ______
ADDRESS ______

VISA

COMPUTER	ITEM	PRICE
		£
		£
		£

Ouside Europe add £1 for each item airmail £ ____ TOTAL £ ____

Send me the FREE Tasman brochure describing your products. tick here: ☐
I would like to know more about your programs for:
ZX Spectrum ☐ MSX ☐ Amstrad CPC 464 ☐

SINCLAIR C5 CLIVEMOBILE
Competition
RESULTS

MODEL T FORD, Volkswagen Beetle, E-Type Jaguar, Austin Mini, Rolls Royce. . . Sinclair C5? Perhaps not. Sinclair's electric vehicle seems unlike to take its place in the pantheon of automative greats. It has already received a buffeting at the hands of the national press which has gleefully reported its mishaps — from the first case of drunken driving on a C5 to the first accident on a C5.

But the company can take comfort in the fact that almost 2,000 *Your Computer* readers would like to own a C5. That is how many people entered our February competition which offered the supertrike as a prize. We asked them to place five features in order of preference and then to invent a computerised gadget which would make driving easier or more fun. Our panel's perference were as follows: 1. Low purchase price; 2. Low running costs, 3. Can be driven by 14-year olds; 4. Easy to park; 5. Lightweight.

The great majority of the entries made serious sounding suggestions. They included radar devices to warn of other cars, satellite-based navigation systems, gadgets to assist parking, devices to monitor the state of the vehicle, and a variety of VDU units — P. Kolenbrander's Super Guide consists of a flat screen display linked to database containing information on Britain's motorways. Its purpose is to direct the driver along the quickest and most economical route.

Other entries were aimed mainly at the C5 rather than motoring in general and took a lighter and more irreverent approach. A. Hartley's gadget — named the Pinking Engine — is a programmable squeaker cum knocker and muffled sound generator, designed to restore character to the vehicle. K. Ciscombe suggested a pump for bailing out the C5 when it rains. There were also several mother-in-law jokes which we mention here only to express our disapproval — The Worsley Patent, for example, feeds back mother-in-law's voice, anti-phase, to the back seat.

Best of all, we thought, was the Roadbot from P. Carrahar Gwynfa, Llanarmon Rd, Llanferres Mold, Clwyd, which wins him the prize. Roadbot takes us back to the days around the turn of the century when cars were restricted to a fast walking pace. It is robot which walks in front of the C5 waving a red flag.



TOP ★ TIPS

REM-LITE

Ever written a program on a Commodore 64, filled it with Rem statements for easy reading and then tried to find the relevant statement as it flashes by on listing?

Sure, you can surround your remarks by lines of asterisks, but what an unnecessary waste. Here is a Basic program which will generate some machine code to sit in memory, away from Basic, and solve this problem.

Having loaded and run this program, then any Basic program may be loaded, and by typing SYS 50000 all of the Rem statements will be inverted. Try listing it and see!! The program may then be saved as normal. Its Rem statements will always show as inverted whenever the program is subsequently loaded and listed.

Richard Lievers

```
5 PRINT CHR$(147)"SYS 50000
  HIGHLIGHTS REM STATEMENTS"
10 DATA 165,43,133,98,165
20 DATA 44,133,99,165,98
30 DATA 208,4,165,99,240
40 DATA 39,160,3,200,177
50 DATA 98,201,0,240,13
60 DATA 201,143,208,245,200
70 DATA 177,98,240,4,169
80 DATA 18,145,98,160,0
90 DATA 177,98,72,200,177
100 DATA 98,133,99,104,133
110 DATA 98,169,1,208,209,96
120 FORN=0TO55:READA:POKE50000+N,A:NEXT
```

LINE DELETE

The following machine-code program for the Spectrum makes the job of deleting a block of Basic program lines almost instantaneous. To use this facility simply merge lines 9000 to 9090 with the program you want to trim and type GOTO 9000. Alternatively you could save the machine code itself with

SAVE "DELETE" CODE 32000,53

and then type in lines 9030 to 9080.

The code is relocatable and can be loaded into any other part of Ram by changing line 9005. As it stands Clear 31999 should be entered as a direct command before the program is run.

Rodney Francis

```
9000 LET t=0
9005 FOR n=32000 TO 32052
9010 READ a: POKE n,a: LET t=t+a
9015 NEXT n
9020 IF t<>[illegible] THEN PRINT "DATA
 ERROR": STOP
9030 INPUT "First line";f
9040 INPUT "Last line";l
9050 POKE 23296,f-256*INT (f/256
): POKE 23297,INT (f/256)
9060 POKE 23298,l-256*INT (l/256
): POKE 23299,INT (l/256)
9070 RANDOMIZE USR 32000
9080 DATA [illegible]
9090 DATA [illegible]
```

RESPONSE *Frame*

Do you have a problem related to your micro? Tim Hartnell will do his best to help. Please include only one question per letter and mark it "Response Frame". Alternatively, perhaps you have an idea you'd like to pass on to others. Why not write to us with your top tips?

SPRITE CONTROL

■I have recently acquired a Commodore 64, and have been writing programs involving sprites. I've come up against the problem of being able to control the sprite by using the keyboard. I want to press, for example, the Z key to move left and the X to move right.

R King,
Rochester, Kent.

THIS IS FAIRLY easy to do. You read the keyboard with Get$, and then Poke the relevant character and colour locations of your object with their addresses plus one, to move right, and minus one to move left. If char was the current character address, and col was the current colour address, your program could look like this:

```
10 GET A$
20 IF A$<>"Z" AND A$<>"X"
   THEN 10
30 IF A$="Z" THEN POKE char
   -1,n:POKE col-1,nn
40 IF A$="X" THEN POKE char
   +1,n:POKE col+1,nn
50 GOTO 10
```

NO PRINT

■I have had a Vic-20 for some time, and have written a number of programs for it. I have recently bought a 16K expansion, and found that the games which included screen memory mapping do not print anything where a screen character is supposed to be. Why is this, and how can I fix it?

N G Watkin,
Dereham, Norfolk.

WHEN YOU ADD 8K or more to the Vic, the screen locations change. The character memory map moves from 7680 to 4096. Therefore, to get your old programs to run, you need to subtract 3584 from each character Poke to the screen. Colour Pokes do not change.

ZX-81 DATABASE

■I have a 16K ZX-81 and would like to be able to keep records of my bird's breeding sessions on it. I don't really know a lot about programming, and would appreciate it if you could suggest a way of storing my records on computer. I have enclosed a record card showing the information I keep. I have 20 pairs of birds.

Ralph Martin,
Gillingham, Kent.

THE IMPORTANT question to settle, when trying to set up a database, is what you want to be able to get out of it. This is not meant to be facetious. I assume you would need to store information so it could be accessed by the identity of its parents, so you could ask, in effect, questions like "How many eggs have birds A and B laid this year?", and then go on to find out such things as how many female chicks were born, and so on. A database is essentially, a multidimensional array, which can be cross-referenced. Your prime arrays could hold the identity of the parents, with the other information you need tagged to these. The subject is too vast for me to be able to answer adequately in a column like this one, but I can point you in the right direction. Three books on databases which I recommend are *The Database Primer*, Rose Deakin (Century, 1983); *Database, A Primer*, C J Date (Addison-Wesley, 1983); and the rather oddly titled *Databases for Fun and Profit*, Nigel Freestone (Granada, 1983). I hope these can help.

SHARP MOVES

■I am the owner of a Sharp MZ-80K and am interested in making up my own games. I can't find out how to do such things as move a base cannon, fire a laser, and move alien ships. I would be very pleased if you could help me.

Nigel Russell,
Beckingham,
Doncaster.

THE SHARP has little arrow characters which you can include in Print statements to locate objects at particular positions on the screen, and move them around. The simplest way to achieve the movement you want would be to go to a subroutine which prints in the arrows, ending with a semicolon. So the subroutine includes, say,

```
FOR J = 1 TO 10:PRINT
  "arrow-right";:NEXT J
```

before returning to the main program where the actual character is printed. If the end of the loop — the 10 in the example line — was a variable, then you could assign it before calling the subroutine, and thus get it to move — in this case — the character as many positions to the right as you want. You can emulate a firing laser by moving a full stop, or a dash, across the screen, again with the embedded arrow characters.

BREAK-PROOFING

■I own an Atari 400. I am writing a program in which a password must be entered to run another program. As the program is written in Atari Basic, it is easy for someone to list the program, and discover the password. Could you please tell me how to prevent the listing of the program?

Philip Chandler,
Mossley Hill, Liverpool.

THERE IS NO way, in Basic, to force an auto-run on the Atari machines. However you can ensure that — once running — the program can't be listed by adding the line

```
POKE 16,64:POKE 53774,64
```

into the program. This must appear in the program after each and every use of a Graphics call. I suggest you hide the code in some way, perhaps by checking half of the ASCII code of each letter, and adding a value to it which comes from somewhere else (such as hidden within a long set of data). Make it so that the password cannot be instantly spotted when the program is listed, and you'll have a good chance of ensuring it is secure.

ATARI MERGE

■I have an Atari 600XL and am very happy with it. Unfortunately, I cannot make it merge a program from tape, with one which is already in the computer's memory.

A S Dewdney,
Durham.

SIMPLY SAVE the second program listing with List "C", filename for cassette or List "D", filename for disc. Then, get the first program back in the computer, and load in the second one with the command Enter "C", or "D", filename. This should do the trick.

"Macbeth, THE Adventure of the Year"

Personal Computer News, November '84

- ***"An extremely well-presented adventure with highly atmospheric graphics."*** Observer, December '84
- ***"Macbeth is something special."*** Personal Computer World, December '84
- ***"Macbeth ranks as one of the most varied, inventive and original computer games."*** Times Literary Supplement, December '84
- ***"Exciting graphics and sound are the highlights of these masterpieces."*** Commodore Computing, January '85
- ***"Contains no less than four separate adventures for £14.95, excellent value."*** Daily Express, December '84
- ***"Macbeth is superb – a programming tour de force."*** Micro Adventurer, February '85

Macbeth THE Computer Adventure

For Commodore 64

FROM ALL GOOD SOFTWARE
STOCKISTS, OR DIRECT.

CREATIVE SPARKS

Department MO, Thomson House, 296 Farnborough Road, Farnborough, Hants. GU14 7NU.

SOFTWARE *File*

Programs for Software File should be compact and sent on a cassette. Please include clear instructions and say what computer it's for. We pay between £6 and £36 for programs published. They must be double-checked and submitted to *Your Computer* exclusively.

Kung Fu

Michael Sutherland, Aberdeen.

SPECTRUM

KUNG FU IS A game for two players on the 48K ZX Spectrum. It is based on the age-old martial art of Kung Fu. Each player controls his own character which can kick, punch and duck as well as move left and right.

You can only punch your opponent when he is right next to you, but when he is further away you can send him skidding over the sand-pit on his back with a flying kick. If you knock your opponent out of the sand-pit you cannot hit him further and he gets a chance to clamber back in to the arena.

Each time the character is hit he loses a hit point until eventually he is defeated when his hit points are reduced to zero. The control keys for each player are given in the program.

Type in the program as listed and secure on tape with

SAVE "KUNG FU" LINE 1

For any reader who does not wish to type in the program, a cassette of it may be obtained by sending your name and address together with a cheque for £1.50 to Michael Sutherland, 23 Forest Avenue, Abe.deen AB1 6TU.

```
  5 REM ++ © Mike Sutherland ++
  7 REM +++++++++ 1985 +++++++++
 10 REM +++ INITIALISATION +++
 11 BORDER 2: PAPER 5: INK 0: C
LS
 12 PRINT AT 3,4;"PLEASE WAIT-P
OKING DATA": GO TO 9900
 15 PRINT AT 0,1; FLASH 1;"■>■>
■>■>■>■"; FLASH 0;" KUNG FU "; F
LASH 1;"■/■/■/■/■/■": RETURN
  5 REM ++ © Mike Sutherland ++
  7 REM +++++++++ 1985 +++++++++
 10 REM +++ INITIALISATION +++
 11 BORDER 2: PAPER 5: INK 0: C
LS
 12 PRINT AT 0,4;"PLEASE WAIT-P
OKING DATA": GO TO 9900
 15 PRINT AT 0,1; FLASH 1;"■>■>
■>■>■>■"; FLASH 0;" KUNG FU "; F
LASH 1;"■<■<■<■<■<■": RETURN
 17 GO SUB 15: PRINT AT 14,5;"V
IEW CONTROL KEYS (Y/N)": IF INKE
Y$="Y" OR INKEY$="y" THEN GO TO
9000
 19 IF INKEY$="N" OR INKEY$="n"
 THEN PRINT AT 14,5;"
                ": GO TO 22
 21 GO TO 17
 22 PRINT AT 5,0;"BLUE PLAYER,"
;AT 10,0;"ENTER CHARACTER'S NAME
:        (max.14 chars.)": POKE
 23617,236: POKE 23658,8: INPUT
LINE g$: IF LEN g$>14 THEN GO TO
 22
 24 PRINT AT 5,0;"RED PLAYER, "
: INPUT LINE h$: IF LEN h$>14 TH
EN GO TO 24
 40 REM ++++ SET VARIABLES ++++
 41 LET hb=5: LET vb=20
 42 LET hr=25: LET vr=20
 43 LET hpb=20: LET hpr=20
 45 LET a$="P": LET b$="I"
 46 LET c$="H": LET d$="A"
 48 LET xv=0: LET yv=0
 50 REM ++++ SET UP SCREEN ++++
 60 BORDER 0: PAPER 6: INK 7: C
LS
 65 FOR f=0 TO 255: PRINT INK I
NT (RND+8);"G";: NEXT f
 70 FOR f=0 TO 95: PRINT PAPER
1;" ";: NEXT f: FOR f=0 TO 95: P
RINT PAPER 2;" ";: NEXT f
 75 PRINT AT 9,1; INK 7; PAPER
1;g$;AT 9,16;"HIT POINTS : ";hpb
;" ";AT 12,1; PAPER 2;h$;AT 12,1
6;"HIT POINTS : ";hpr;" "
 80 FOR f=3 TO 27: PRINT AT 21,
f; INK 0;"O": NEXT f
 85 PRINT AT 21,0; PAPER 0;"
";AT 21,28;"
 87 FOR g=1 TO 4: FOR f=1 TO 4:
 PRINT AT 16,10; INK 0;"BEGIN CO
MBAT": BEEP .05,20: BEEP .05,10:
 NEXT f: PRINT AT 16,10; INK 7;"
              ": FOR h=1 TO 12: NE
XT h: NEXT g
 90 IF yv=0 THEN PRINT AT vb,hb
; INK 0;a$;AT vb-1,hb; INK 1;b$
 95 IF xv=0 THEN PRINT AT vr,hr
; INK 0;c$;AT vr-1,hr; INK 2;d$
100 GO TO 4000
500 REM +++++ MOVE L B +++++
510 LET hb=hb-(hb>3): IF (hb>3)
=0 THEN GO TO 530
520 PRINT AT vb,hb; INK 0;a$;AT
 vb-1,hb; INK 1;b$;AT vb,hb+1; I
NK 7;" ";AT vb-1,hb+1;" "
530 PRINT AT vb,hb+1; INK 7;" "
;AT vb-1,hb+1;" ": RETURN
600 REM +++++ MOVE R B +++++
610 LET hb=hb+(hb<27): IF (hb<2
7)=0 THEN GO TO 630
620 PRINT AT vb,hb; INK 0;a$;AT
 vb-1,hb; INK 1;b$;AT vb,hb-1; I
NK 7;" ";AT vb-1,hb-1;" "
630 PRINT AT vb,hb-1; INK 7;" "
;AT vb-1,hb-1;" ": RETURN
700 REM +++++ MOVE L R +++++
710 LET hr=hr-(hr>3): IF (hr>3)
=0 THEN GO TO 730
720 PRINT AT vr,hr; INK 0;c$;AT
 vr-1,hr; INK 2;d$;AT vr,hr+1; I
NK 7;" ";AT vr-1,hr+1;" "
730 PRINT AT vr,hr+1; INK 7;" "
;AT vr-1,hr+1;" ": RETURN
800 REM +++++ MOVE R R +++++
810 LET hr=hr+(hr<27): IF (hr<2
7)=0 THEN GO TO 830
820 PRINT AT vr,hr; INK 0;c$;AT
 vr-1,hr; INK 2;d$;AT vr,hr-1; I
NK 7;" ";AT vr-1,hr-1;" "
830 PRINT AT vr,hr-1; INK 7;" "
;AT vr-1,hr-1;" ": RETURN
1000 REM +++++ BLUE DEAD +++++
1010 PRINT AT 16,3; INK 0;g$;AT
16,18;"IS DEFEATED";AT 17,3;h$;A
T 17,18;"IS VICTORIOUS"
1020 GO TO 1270
1250 REM +++++ RED DEAD +++++
1260 PRINT AT 16,3; INK 0;h$;AT
16,18;"IS DEFEATED";AT 17,3;g$;A
T 17,18;"IS VICTORIOUS"
1265 REM ++++ KEY SCAN ++++
1270 PRINT AT 21,6; INK 0;"ANOTH
ER GAME ? (Y/N)"
1280 FOR f=10 TO 30: BEEP .01,f
1285 IF INKEY$="Y" OR INKEY$="y"
 THEN GO TO 40
1290 IF INKEY$="N" OR INKEY$="n"
 THEN STOP
1295 NEXT f
1300 GO TO 1280
1500 REM +++++ PUNCH B +++++
1510 IF b$="A" THEN PRINT AT vb-
1,hb; INK 1;"C": IF ATTR (vb-1,h
b-1)=50 THEN LET hpr=hpr-1: LET
hr=hr-(2 AND hr>3): PRINT AT vr,
hr; INK 2;"F";AT vr,hr+1; INK 0;
"E";AT vr,hr+2; INK 7;" ";AT vr-
1,hr+2;" ";AT vr-1,hr;" ": BEEP
.1,20: BEEP .1,10: GO SUB 3100:
PRINT AT vr,hr; INK 0;c$;AT vr,h
r+1; INK 7;" ";AT vr-1,hr; INK 2
;d$: GO TO 1530
1520 IF b$="I" THEN PRINT AT vb-
1,hb; INK 1;"L": IF ATTR (vb-1,h
b+1)=50 THEN LET hpr=hpr-1: LET
hr=hr+(2 AND hr<27): PRINT AT vr
,hr; INK 2;"N";AT vr,hr-1; INK 0
;"M";AT vr,hr-2; INK 7;" ";AT vr
-1,hr-2;" ";AT vr-1,hr;" ": BEEP
 .1,20: BEEP .1,10: GO SUB 3100:
 PRINT AT vr,hr; INK 0;c$;AT vr,
hr-1; INK 7;" ";AT vr-1,hr; INK
2;d$: GO TO 1530
1530 IF hpr=0 THEN GO TO 1250
1540 RETURN
1750 REM +++++ PUNCH R +++++
1760 IF d$="A" THEN PRINT AT vr-
1,hr; INK 2;"D": IF ATTR (vr-1,h
r-1)=49 THEN LET hpb=hpb-1: LET
hb=hb-(2 AND hb>3): PRINT AT vb,
hb; INK 1;"F";AT vb,hb+1; INK 0;
"E";AT vb,hb+2; INK 7;" ";AT vb-
1,hb+2;" ";AT vb-1,hb;" ": BEEP
.1,20: BEEP .1,10: GO SUB 3000:
PRINT AT vb,hb+1; INK 7;" ";AT v
b,hb; INK 0;a$;AT vb-1,hb; INK 1
;b$: GO TO 1780
1770 IF d$="I" THEN PRINT AT vr-
1,hr; INK 2;"L": IF ATTR (vr-1,h
r+1)=49 THEN LET hpb=hpb-1: LET
hb=hb+(2 AND hb<27): PRINT AT vb
,hb; INK 1;"N";AT vb,hb-1; INK 0
;"M";AT vb,hb-2; INK 7;" ";AT vb
-1,hb-2;" ";AT vb-1,hb;" ": BEEP
 .1,20: BEEP .1,10: GO SUB 3000:
 PRINT AT vb,hb-1; INK 7;" ";AT
vb,hb; INK 0;a$;AT vb-1,hb; INK
1;b$: GO TO 1780
1780 IF hpb=0 THEN GO TO 1000
1790 RETURN
2000 REM +++++ DUCK B +++++
2010 PRINT AT vb,hb; INK 1;b$;AT
 vb-1,hb; INK 7;" "
2020 RETURN
2250 REM +++++ DUCK R +++++
2260 PRINT AT vr,hr; INK 2;d$;AT
 vr-1,hr; INK 7;" "
2270 RETURN
2500 REM +++++ KICK B +++++
2510 IF b$="A" THEN PRINT AT vb,
hb; INK 7;" ";AT vb-1,hb; INK 1;
"C";AT vb-1,hb-1; INK 0;"B": LET
 no=0: GO TO 2540
2520 IF b$="I" THEN PRINT AT vb,
hb; INK 7;" ";AT vb-1,hb; INK 1;
"K";AT vb-1,hb+1; INK 0;"J": LET
 no=0: GO TO 2550
2530 IF hpr=0 THEN GO TO 1250
2535 RETURN
2540 IF hb-1<6 THEN PRINT AT vb,
hb; INK 0;a$;AT vb-1,hb; INK 1;b
$;AT vb-1,hb-1; INK 7;" ": RETUR
N
2542 PRINT AT vb-1,hb-1; INK 0;"
B";AT vb-1,hb; INK 1;"C";AT vb-1
,hb+1; INK 7;" ": IF ATTR (vb-1,
hb-2)=50 THEN LET hpr=hpr-1: BEE
P .1,20: BEEP .1,10: LET hb=hb-(
hb>3): PRINT AT vb,hb; INK 0;a$;
AT vb-1,hb; INK 1;b$;AT vb-1,hb+
1; INK 7;" ": PRINT AT vr,hr; IN
K 0;"E";AT vr,hr-1; INK 2;"F";AT
 vr-1,hr; INK 7;" ": FOR f=1 TO
4: NEXT f: PRINT AT vr,hr-1; INK
 0;"E";AT vr,hr-2; INK 2;"F";AT
vr,hr; INK 7;" ": FOR f=1 TO 8:
NEXT f: PRINT AT vr,hr-2; INK 0;
"E";AT vr,hr-3; INK 2;"F";AT vr,
hr-1; INK 7;" ": GO SUB 3100: LE
T hr=hr-3: FOR f=1 TO 20: NEXT f
: PRINT AT vr,hr; INK 0;c$;AT vr
-1,hr; INK 2;d$;AT vr,hr+1; INK
7;" ": GO TO 2530
2544 LET no=no+1: IF no=3 THEN L
ET hb=hb-1: GO SUB 2530: PRINT A
T vb,hb; INK 0;a$;AT vb-1,hb; IN
K 1;b$;AT vb-1,hb+1; INK 7;" ":
RETURN
2546 LET hb=hb-1: FOR f=1 TO 4:
NEXT f: GO TO 2542
2550 IF hb+1>24 THEN PRINT AT vb
,hb; INK 0;a$;AT vb-1,hb; INK 1;
b$;AT vb-1,hb+1; INK 7;" ": RETU
RN
2552 PRINT AT vb-1,hb; INK 1;"K"
;AT vb-1,hb+1; INK 0;"J";AT vb-1
,hb-1; INK 7;" ": IF ATTR (vb-1,
hb+2)=50 THEN LET hpr=hpr-1: BEE
P .1,20: BEEP .1,10: LET hb=hb+(
hb<27): PRINT AT vb,hb; INK 0;a$
;AT vb-1,hb; INK 1;b$;AT vb-1,hb
-1; INK 7;" ": PRINT AT vr,hr; I
NK 0;"M";AT vr,hr+1; INK 2;"N";A
T vr-1,hr; INK 7;" ": FOR f=1 TO
 4: NEXT f: PRINT AT vr,hr+1; IN
K 0;"M";AT vr,hr+2; INK 2;"N";AT
 vr,hr; INK 7;" ": FOR f=1 TO 8:
 NEXT f: PRINT AT vr,hr+2; INK 0
;"M";AT vr,hr+3; INK 2;"N";AT vr
,hr+1; INK 7;" ": GO SUB 3100: L
ET hr=hr+3: FOR f=1 TO 20: NEXT
f: PRINT AT vr,hr; INK 0;c$;AT v
r-1,hr; INK 2;d$;AT vr,hr-1; INK
 7;" ";AT vr,hr+1;" ": GO TO 253
0
2554 LET no=no+1: IF no=3 THEN L
ET hb=hb+1: GO SUB 2530: PRINT A
T vb,hb; INK 0;a$;AT vb-1,hb; IN
K 1;b$;AT vb-1,hb-1; INK 7;" ":
RETURN
2556 LET hb=hb+1: FOR f=1 TO 4:
NEXT f: GO TO 2552
2750 REM +++++ KICK R +++++
2760 IF d$="A" THEN PRINT AT vr,
hr; INK 7;" ";AT vr-1,hr; INK 2;
"C";AT vr-1,hr-1; INK 0;"B": LET
 no=0: GO TO 2790
2770 IF d$="I" THEN PRINT AT vr,
hr; INK 7;" ";AT vr-1,hr; INK 2;
```

(continued on next page)

(continued from previous page)

```
"K";AT vr-1,hr+1; INK 0;"J": LET
 no=0: GO TO 2800
2780 IF hpb=0 THEN GO TO 1000
2785 RETURN
2790 IF hr-1<6 THEN PRINT AT vr,
hr; INK 0;c$;AT vr-1,hr; INK 2;d
$;AT vr-1,hr-1; INK 7;" ": RETUR
N
2792 PRINT AT vr-1,hr; INK 2;"C"
;AT vr-1,hr-1; INK 0;"B";AT vr-1
,hr+1; INK 7;" ": IF ATTR (vr-1,
hr-2)=49 THEN LET hpb=hpb-1: BEE
P .1,20: BEEP .1,10: LET hr=hr-(
hr>3): PRINT AT vr,hr; INK 0;c$;
AT vr-1,hr; INK 2;d$;AT vr-1,hr+
1; INK 7;" ": PRINT AT vb,hb; IN
K 0;"E";AT vb,hb-1; INK 1;"F";AT
 vb-1,hb; INK 7;" ": FOR f=1 TO
4: NEXT f: PRINT AT vb,hb-1; INK
 0;"E";AT vb,hb-2; INK 1;"F";AT
vb,hb; INK 7;" ": FOR f=1 TO 8:
NEXT f: PRINT AT vb,hb-2; INK 0;
"E";AT vb,hb-3; INK 1;"F";AT vb,
hb-1; INK 7;" ": GO SUB 3000: LE
T hb=hb-3: FOR f=1 TO 20: NEXT f
: PRINT AT vb,hb; INK 0;a$;AT vb
-1,hb; INK 1;b$;AT vb,hb+1; INK
7;" ": GO TO 2780
2794 LET no=no+1: IF no=3 THEN L
ET hr=hr-1: GO SUB 2780: PRINT A
T vr,hr; INK 0;c$;AT vr-1,hr; IN
K 2;d$;AT vr-1,hr+1; INK 7;" ":
RETURN
2796 LET hr=hr-1: FOR f=1 TO 4:
NEXT f: GO TO 2792
2800 IF hr+1>24 THEN PRINT AT vr
,hr; INK 0;c$;AT vr-1,hr; INK 2;
d$;AT vr-1,hr+1; INK 7;" ": RETU
RN
2802 PRINT AT vr-1,hr; INK 2;"K"
;AT vr-1,hr+1; INK 0;"J";AT vr-1
,hr-1; INK 7;" ": IF ATTR (vr-1,
hr+2)=49 THEN LET hpb=hpb-1: BEE
P .1,20: BEEP .1,10: LET hr=hr+(
hr<27): PRINT AT vr,hr; INK 0;c$
;AT vr-1,hr; INK 2;d$;AT vr-1,hr
-1; INK 7;" ": PRINT AT vb,hb; I
NK 0;"M";AT vb,hb+1; INK 1;"N";A
T vb-1,hb; INK 7;" ": FOR f=1 TO
 4: NEXT f: PRINT AT vb,hb+1; IN
K 0;"M";AT vb,hb+2; INK 1;"N";AT
 vb,hb; INK 7;" ": FOR f=1 TO 8:
 NEXT f: PRINT AT vb,hb+2; INK 0
;"M";AT vb,hb+3; INK 1;"N";AT vb
,hb+1; INK 7;" ": GO SUB 3000: L
ET hb=hb+3: FOR f=1 TO 20: NEXT
f: PRINT AT vb,hb; INK 0;a$;AT v
b-1,hb; INK 1;b$;AT vb,hb-1; INK
 7;" ": GO TO 2780
2804 LET no=no+1: IF no=3 THEN L
ET hr=hr+1: PRINT AT vr,hr; INK
0;c$;AT vr-1,hr; INK 2;d$;AT vr-
1,hr-1; INK 7;" ": RETURN
2806 LET hr=hr+1: GO TO 2802
3000 REM *** PRINT B SCORE ***
3010 PRINT AT 9,29; INK 7; PAPER
 1;hpb;" ": RETURN
3100 REM *** PRINT R SCORE ***
3110 PRINT AT 12,29; INK 7; PAPE
R 2;hpr;" ": RETURN
4000 REM ***** KEYBOARD *****
4005 LET xv=0: LET yv=0
4010 LET a=IN 63486: LET b=IN 61
438
4015 LET k=IN 65022
4020 LET c=IN 64510: LET d=IN 57
342
4030 LET e=IN 65278: LET f=IN 49
150
4038 IF b=k-1 THEN GO SUB 2250:
LET xv=1
4040 IF a=k-1 THEN GO SUB 2000:
LET yv=1: GO TO 4060
4042 IF c=k-2 THEN LET a$="H": L
ET b$="A": GO SUB 500
4044 IF c=k-4 THEN LET a$="P": L
ET b$="I": GO SUB 600
4046 IF e=k-1 THEN GO SUB 2500
4048 IF e=k-2 THEN GO SUB 1500
4060 IF xv=1 THEN GO TO 4140
4070 IF d=k-8 THEN LET c$="H": L
ET d$="A": GO SUB 700
4090 IF d=k-4 THEN LET c$="P": L
ET d$="I": GO SUB 800
4110 IF f=k-1 THEN GO SUB 2750
4130 IF f=k-2 THEN GO SUB 1750
4140 GO TO 90
9000 CLS : GO SUB 15: PRINT AT 1
,8;">> CONTROL KEYS <<";AT 3,8;"
BLUE player  RED player";AT 5,1;
"LEFT  :         U             U";AT 7
,0;"RIGHT :         E              I";
AT 9,1;"DUCK  :         1
0"
9010 PRINT AT 11,1;"KICK  :     C/S
HIFT           ENTER";AT 13,0;"PUNCH
 :        Z               L"
9020 POKE 23659,0: PRINT AT 22,3
; INK 2; PAPER 7;"PRESS ANY KEY
TO CONTINUE": POKE 23659,2
9030 IF INKEY$="" THEN GO TO 903
0
9040 CLS : GO SUB 15: GO TO 22
9900 REM ***** U.D.G.'s *****
9905 RESTORE 9920
9910 FOR f=USR "a" TO USR "p"+7:
 READ a: POKE f,a: NEXT f
9915 GO TO 17
9920 DATA 12,30,12,30,26,26,26,2
2,0,0,128,255,3,9,15,0,6,15,6,25
4,190,196,252,0,12,30,12,158,250
,26,26,22
9930 DATA 0,0,1,15,248,193,207,2
48,0,96,240,96,127,125,35,63,56,
84,108,124,56,124,254,254,30,30,
54,98,67,65,195,0
9940 DATA 48,120,48,120,88,88,88
,104,0,0,1,255,192,144,240,0,96,
240,96,127,125,35,63,0,48,120,48
,121,95,88,88,104
9950 DATA 0,0,128,240,31,131,243
,31,0,6,15,6,254,190,196,252,132
,32,73,4,48,138,36,255,120,120,1
08,70,194,130,195,0
```

Backpack Bozo

J Powell and J Burman,
Cape Town,
South Africa.

BBC

BACKPACK BOZO is a game for the BBC Micro Model B or Model A with 32K. It is mostly written in BBC Basic with a small section of machine code. Due to the advanced nature of the BBC Basic it will be difficult to convert to any other computer.

The aim of the game is to fly Backpack Bozo around the screen squashing the mutated scorpions and landing on the bonus fuel tanks. Your fuel will decrease as you jet around the screen. When your score reaches 4,000, the gravity will double and you will get an extra man. It is advisable not to land on the ground as it is highly acidic and will eat through your suit. As the game progresses the scorpions will move faster and faster.

Before you enter the program, type in the following line

*KEYO|!u

Whenever you see an italic "u" in lines 350, 370 and 1370, press the red function key f0.

```
   50 MODE7
   60 VDU23;8202;0;0;0;
   70 IF A%=0 THEN PROCTITLE
   80 FUEL=1279
   90 X=0
  100 DIR=0:ODIR=0
  110 SP=0
  120 PROCASS
  130 POD=0
  140 LI=3
  150 SPP=0
  160 SPIS=5
  170 GRAV=1
  180 *FX10,2
  190 *FX9,2
  200 *TV255,1
  210 MODE5
  220 COLOUR1
  230 PRINTTAB(0,0);"LIVES 3   SCORE "
  240 SC=0
  250 *FX11,1
  260 *FX12,1
  270 VDU23,240,96,176,188,116,124,124,252,112
  280 VDU23,241,112,112,120,124,108,102,102,238
  290 VDU23,242,6,13,61,46,62,62,63,14
  300 VDU23,243,14,14,30,62,54,102,102,119
  310 VDU23,244,0,0,0,192,96,62,126,165
  320 VDU23,245,255,255,255,255,255,255,255,255
  330 VDU23;8202;0;0;0;
  340 COLOUR3
  350 PRINTTAB(0,24);"uuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuu
uuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuu
uuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuuu"
  360 COLOUR2
  370 PRINTTAB(0,26);"uuuuuuuuuuuuuuuuuuuu"
  380 VDU19,2,2,0,0,0
  390 HI=975
  400 VEL=0
  410 X=600
  420 HI2=HI
  430 X2=X
  440 VDU5
  450 GCOL3,2
  460 MOVE X,HI:VDU240+DIR:MOVEX,HI-32:VDU241+DIR
  470 PROCMO
  480 PROCCOMPUTE
  490 IF FUEL>0PROCKEYP ELSE *FX15,0
  500 PROCFUEL
  510 PROCSPID
  520 PROCPOD
  530 IF SC>=4000 AND GRAV=1 THEN PROCGRAV
  540 GOTO470
  550 DEFPROCMO
  560 GCOL3,2
  570 ?&810=X2 MOD 256:?&815=X2 DIV256:?&81A=HI2 MOD 2
56:?&81F=HI2 DIV 256:?&833=X2 MOD 256:?&838=X2 DIV256:
?&83D=(HI2-32) MOD 256:?&842=(HI2-32) DIV 256
  580 ?&824=240+ODIR:?&847=241+ODIR:?&86A=240+DIR:?&88
D=241+DIR
  590 ?&856=X MOD 256:?&85B=X DIV256:?&860=HI MOD 256:
?&865=HI DIV 256:?&879=X MOD 256:?&87E=X DIV256:?&883=
(HI-32) MOD 256:?&888=(HI-32) DIV 256
  600 CALL&800
  610 X2=X:HI2=HI
  620 ODIR=DIR
  630 IF HI<320 THEN PROCDIE:GOTO390
  640 IF POINT(X,HI-64)=1 OR POINT(X+60,HI-64)=1 OR PO
INT(X+30,HI-64)=1 THEN PROCSQUA
  650 ENDPROC
  660 DEFPROCCOMPUTE
  670 VEL=VEL-GRAV
  680 IF VEL<=-15 THEN VEL=-15
  690 IF VEL>=15 THEN VEL=15
  700 HI=HI+VEL
  710 IF HI>975 THEN HI=975:VEL=0
  720 ENDPROC
  730 DEFPROCKEYP
  740 IF INKEY(-98) THEN X=X-20:FUEL=FUEL-8:DIR=0:GOTO
760
  750 IF INKEY(-67) THEN X=X+20:FUEL=FUEL-8:DIR=2
  760 IF INKEY(-74) THEN VEL=VEL+4:SOUND0,-8,6,4:FUEL=
FUEL-8 ELSE *FX15,0
  770 IF X<0 THEN X=0
  780 IF X>1210 THEN X=1210
  790 ENDPROC
  800 DEFPROCASS
  810 P%=&800
  820 [OPT0:LDA£19:JSR&FFF4
  830 ]
  840 FORN=1 TO 28
  850 READ A
  860 [OPT0:LDA£A:JSR&FFEE
  870 ]
  880 NEXTN
  890 [OPT0:RTS
  900 ]
  910 ENDPROC
  920 DATA 25,4,88,2,255,3,240,25,4,88,2,223,3,241,25,
4,88,2,255,3,240,25,4,88,2,223,3,241
  930 DEFPROCSPID
  940 GCOL3,1
  950 IF SP=1 THEN 1000
  960 IF RND(15)<>3 THEN ENDPROC
  970 SP=1
  980 SX=1260
  990 SOX=SX:MOVESX,205:VDU244
 1000 SX=SX-SPIS
 1010 SPIS=5+(SC-SPP)/100
 1020 IF SPIS>40 THEN SPIS=40
 1030 MOVESOX,205:VDU244
 1040 MOVESX,205:VDU244
 1050 SOX=SX
 1060 IF SX<-75 THEN SP=0
 1070 ENDPROC
 1080 DEFPROCSQUA
 1090 IF INT(SX/10)<100 THEN SC=SC+100 ELSE SC=SC+INT(
SX/10)
 1100 VDU4:COLOUR1:PRINTTAB(16,0);SC:VDU5
 1110 SOUND0,-15,5,25
 1120 FORM=1 TO 10
 1130 FORN=1 TO 25:NEXTN
 1140 VDU19,1,4,0,0,0
 1150 FORN=1 TO 25:NEXTN
 1160 VDU19,1,1,0,0,0
 1170 NEXT N
 1180 VDU19,1,1,0,0,0
 1190 GCOL3,1
 1200 MOVESX,205:VDU244
 1210 SP=0
 1220 PROCBONE(128+K)
 1230 ENDPROC
 1240 DEFPROCDIE
 1250 SOUND0,-15,6,25
 1260 VDU19,2,11,0,0,0
 1270 VDU19,0,7,0,0,0
 1280 FORN=1 TO 50:NEXT N
 1290 VDU19,0,0,0,0,0
 1300 FORN=1 TO 2000:NEXTN
 1310 FORN=1 TO 150:NEXT N
 1320 VDU19,2,2,0,0,0
 1330 MOVE X2,HI2:VDU240+DIR:MOVEX2,HI2-32:VDU241+DIR
 1340 LI=LI-1
 1350 IF LI=-1 THEN VDU4:COLOUR2:PRINTTAB(5,10);"GAME
OVER":PRINT:PRINT:COLOUR1:PRINT;"PRESS SPACE BAR FOR
                                           ANOTHER GAME
";GOTO 1390
 1360 VDU4:COLOUR1:PRINTTAB(0,0);"LIVES ";LI
 1370 FUEL=1279:COLOUR2:PRINTTAB(0,26);"uuuuuuuuuuuuuu
uuuuuu":VDU5
 1380 ENDPROC
 1390 A=GET
 1400 IF A=32 THEN RUN ELSE 1390
 1410 DEFPROCFUEL
 1420 IF FUEL<125 THEN SOUND1,-15,255,2
 1430 GCOL0,1
 1440 MOVE FUEL,160:DRAW FUEL,190:MOVE FUEL+4,160:DRAW
FUEL+4,190
 1450 ENDPROC
 1460 DEFPROCBONE(A)
 1470 GCOL0,2
 1480 FORN=FUEL TO FUEL+A
 1490 SOUND2,-10,240,1
 1500 MOVE N,160:DRAWN,190
 1510 *FX15,0
 1520 NEXT
 1530 FUEL=FUEL+A+8
 1540 IF FUEL>1278 THEN FUEL=1278
 1550 ENDPROC
 1560 DEFPROCPOD
 1570 IF SC<100 THEN ENDPROC
 1580 IF POD=1 THEN 1670
 1590 SCC=(INT(SC/100))*100
 1600 IF SCC/500<>INT(SCC/500) THEN ENDPROC
 1610 POD=1
 1620 TIME=0
 1630 POX=RND(1200)+20
 1640 IF ABS(POX-X)<100 THEN 1630
 1650 GCOL3,2:MOVEPOX,204:VDU245:GCOL0,0:MOVEPOX,284;P
RINT;"F"
 1660 ENDPROC
 1670 IF TIME>2000 THEN 1720
 1680 IF HI>349 OR ABS(X-POX)>10 THEN ENDPROC
 1690 IF VEL<=-8 THEN GCOL3,2:PROCDIE:GOTO390
 1700 N=1279-FUEL
 1710 PROCBONE(N)
 1720 GCOL3,2:MOVEPOX,284:PRINT;"F":MOVEPOX,204:VDU245
 1730 POD=0
 1740 SC=SC+100
 1750 VDU4:COLOUR1:PRINTTAB(16,0);SC:VDU5
 1760 ENDPROC
 1770 DEFPROCTITLE
 1780 A%=1
 1790 PRINT:PRINT:PRINT:PRINT:PRINT
 1800 PRINT;CHR$141;CHR$145;"     B A C K P A C K  B O
Z O"
 1810 PRINT;CHR$141;CHR$146;"     B A C K P A C K  B O
Z O"
 1820 PRINT:PRINT:PRINT;CHR$136;CHR$130;"  (C)1985  J.
```

```
POWELL AND J.BURMAN"
 1030 PRINT:PRINT::PRINT;CHR$13;;"                INSTRUCTION
S(Y/N)"
 1040 A$=GET$:IF A$="N" THEN 1920
 1050 CLS:PRINT;"Fly Backpack Boro and squash the muta
nt scorpions as they crawl across the       bottom of t
he screen.   The further to  the left you squash it, t
he fewer pointsyou get.    Avoid touching  the ground a
s";
 1060 PRINT;"it is highly acidic and will eat throughy
our suit and kill you."
 1070 PRINT:PRINT;"Your fuel supply will decrease as y
ou   jet around the screen.   You get extra  fuel when
 you squash the scorpions.   Inaddition you get a fuel
 tank every 500  points.   If you land slowly on this"
 1080 PRINT;"within 20 seconds your fuel supply will b
e completely replenished.    When you  reach a score o
f 4000 the gravity will  double and you will get an ex
tra man."
 1090 PRINT:PRINT;"CONTROLS: [Z    X]    RETURN""
 1900 PRINT:PRINT:PRINT;"Press a key to start"
 1910 A=GET
 1920 ENDPROC
 1930 DEFPROCGRAV
 1940 FORN=1 TO 250 STEP 10:SOUND1,-15,N,1:SOUND2,-15,
N+1,1:SOUND3,-15,N+2,1:NEXT
 1950 GRAV=2
 1960 SPP=3995
 1970 K=64
 1980 LI=LI+1
 1990 VDU4:COLOUR1:PRINTTAB(0,0);"LIVES ";LI:VDU5
 2000 ENDPROC
```

Life on 64

Michael Morris Franks,
Maida Vale,
London.

LIFE IS ONE of the most famous computer games and certainly is more interesting than space invaders. The idea is to simulate the growth of a cell colony on a large two-dimensional surface which is arranged like a chessboard. Each square touches eight others. At the beginning each square is either full or empty — you can scatter cells all over the board as you wish.

The rules of Life then take over. If a cell has two or three cells touching it, it survives to the next generation. A full cell dies from loneliness if there are less than two cells touching. It also dies from overcrowding if more than four cells touch it. The cells also reproduce — if there is an empty square which has exactly three cells touching it, a cell is born in it.

These simple rules can produce lively and complex patterns. Mathematical zoologists have identified a number of Life animals such as eaters, guns and gliders. Although Basic can be used to generate the patterns it is exceedingly slow and on a Commodore 64 using an array of 38 dots by 23 dots can take over five minutes to generate each generation. This program uses machine code to produce the new generation and can produce 100 generations in under 90 seconds. It also uses multi-colour graphics and sound.

The code looks at each space in turn and decides whether there will be a cell there in the next generation. If there is to be a cell there it sets the colour of the space — by Poking the colour into the colour matrix — with the live cell colour. If not, it sets the colour matrix to the dead cell colour. Once it has looked at all the squares in turn it goes back and puts a dot — Poke code 81 — if there is a live cell colour for that space, and if not it puts a space there — Poke code 32.

The program allows you to choose the live and dead cell colours. It also allows you to either place the cells yourself with the keyboard or joystick, or if you are lazy the computer can do it for you.

The program checks to see if there has been any change from one generation to the next. If the cell pattern has become stable the program will tell you how many cell generations there have been and then allows you either to re-run the program or end it.

The listing for the program has been produced by a printer using a special program which replaces the cursor control characters with a code which should make the program easier to type in, i.e. instead of a reversed heart symbol the program produces CLR which means clear the screen. This is produced by pressing the shift key and the CLR/Home Key.

```
10 TC=1024:CO=54272
20 KB=197:JV=56320:SI=54272
100 GOSUB 1000
110 GOSUB 2000
120 G=0:N=1   :D=2:PRINT"[HOME,YELLOW,SPACE2]HIT '*1' TO STOP    "
125 POKE 54296,15
126 POKE 54278,255:POKE 54273,N:POKE 54276,17
130 SYS 828
140 N=N+D:IF N=1 OR N>244 THEN D=-D
141 POKE 54276,0:POKE 54277,0:POKE 54278,0
150 REM  IFPEEK(2)>0THEN200
160 IF PEEK(KB)=4 THEN 200
170 G=G+1:GOTO 125
200 PRINT"[HOME,DOWN24]STOPPED AFTER";G;"GOS";CHR$(157)CHR$(148)"E";
210 PRINT"[HOME,WHITE,SPACE3,RVS]R[RVOFF]E-RUN OR [RVS]E[RVOFF]ND ?    "
220 GOSUB 3000
230 IF A$="R"THEN 251
240 IF A$<>"E"THEN 220
248 POKE 53281,1:POKE 53280,3
250 PRINT"[CLEAR,DOWN8,RIGHT2,DOWN,BLUE]HOPE YOU ENJOYED IT":END
251 PRINT"[CLEAR,DOWN2,RIGHT,DOWN4,RIGHT4,DOWN,RVS,YELLOW]"TAB(14)"N[RVOFF]
   EW COLOURS"
252  PRINT TAB(19)"[DOWN2,WHITE]OR":PRINT TAB(9)"[DOWN2,RIGHT3,YELLOW,RIGHT2,
   RVS]O[RVOFF]LD COLOURS"
253 GOSUB 3000
254 IF A$="N"THEN 110
255 IF A$="O"THEN GOSUB 2100:GOTO 120
256 GOTO 251
1000 POKE 53281,2:POKE 53280,4:POKE 53282,4
1005 POKE 53270,PEEK(53270)OR 16
1010 PRINT"[CLEAR,DOWN]"TAB(16)"[WHITE]L I F E"
   :PRINT TAB(15)"[YELLOW]----------"
1030 PRINT TAB(10)"[DOWN3]■[MAGENTA,SPACE]LOADING MACHINE CODE"
   :PRINT TAB(15)"[DOWN]PLEASE WAIT"
1040 GOSUB 5000
1050 PRINT"[CLEAR,SPACE,RIGHT,SPACE12,RVS,YELLOW,SPACE]R E A D Y [RVOFF,
   SPACE3]"
1060 PRINT "[DOWN2,WHITE,SPACE5,RIGHT5]HIT ANY KEY TO START"
1070 GOSUB 3000:RETURN
1500 PRINT"[CLEAR,DOWN,YELLOW,SPACE2]USE THESE KEYS TO POSITION THE CELLS"
1510 PRINT TAB(19)"[DOWN,YELLOW]UP[DOWN,LEFT2]I[DOWN,LEFT,WHITE]S"
   :PRINT TAB(13)"[DOWN,YELLOW]LEFT-[WHITE]< >[YELLOW]-RIGHT"
1515 PRINT TAB(19)"[WHITE,DOWN]X[DOWN,LEFT,YELLOW]I[DOWN,LEFT3]DOWN"
1520 PRINT"[DOWN,SPACE11,WHITE]SPACE-BAR [YELLOW]FOR CELL"
1530 PRINT"[DOWN,SPACE10,DOWN]<OR USE A JOY-STICK>"
1540 PRINT"[DOWN,SPACE14,MAGENTA,RVS,SPACE]HIT ANY KEY "
1550 GOSUB 3000:RETURN
2000 PRINT"[CLEAR]":FOR I=0 TO 7:POKE 646,I:PRINT" ●";:NEXT:PRINT
2010 PRINT"[DOWN,YELLOW]LIVE CELL COLOUR"
2020 INPUT"(1-8)";LC:IF LC<1 OR LC>8 THEN PRINT"[UP2]":GOTO 2060
2030 LC=LC-1:POKE 53283,LC:POKE 646,LC:PRINT TAB(11)"●[DOWN2]"
2040 FOR I=0 TO 15:POKE 646,I:PRINT" ●";:NEXT:PRINT
2050 PRINT"[DOWN,YELLOW]'DYING' CELL COLOUR"
2060 INPUT"(1-16)";DC:IF DC<1 OR DC>16 OR DC-1=LC THEN PRINT"[UP2]":GOTO 2060
2065 DC=DC-1:POKE 646,DC:PRINT TAB(11)"●"
2070 PRINT"[DOWN2,RIGHT2,WHITE]OK (Y/N)?":GOSUB 3000
2080 IF A$<>"Y"THEN 2000
2090 POKE 871,LC:POKE 903,LC:POKE 867,DC
2100 PRINT"[CLEAR,DOWN7,WHITE,RVS]"TAB(13);"P[RVOFF]OSITION CELLS"
   :PRINT TAB(18)"[DOWN2,YELLOW]OR"
2105 PRINT TAB(16)"[DOWN2,RVS]R[RVOFF]ANDOM?"
2110 GOSUB 3000
2120 IF A$="P"THEN GOSUB 2500:RETURN
2130 IF A$="R"THEN GOSUB 2800:RETURN
2140 GOTO 2110
2500 PRINT"[DOWN2,WHITE,SPACE]INSTRUCTIONS [YELLOW](Y/N)?":GOSUB 3000
2510 IF A$="Y"THEN GOSUB 1500
2520 PRINT"[CLEAR,WHITE,SPACE]'F7' WHEN FINISHED":            :X=1:Y=1
2530 P=TC+40*Y+X:U=PEEK(P):UC=PEEK(CO+P)
2535 IF P>1983 THEN P=1982
2536 IF P<1063 THEN P=1064
2537 IF UC>32 AND UC<>81 THEN U=32
2538 POKE P,128+U:POKE CO+P,1
2540 K=PEEK(KB)
2542 J1=PEEK(JV)
2550 IF K= 3 THEN GOSUB 2700
2555 IF K=3 THEN POKE P,U:POKE CO+P,UC:RETURN
2560 IF K=60 OR(J1 AND 16)=0  THEN U=81:UC=LC:GOTO 2535
2570 NX=X+(K=47 OR(J1 AND 4)=0)-(K=44 OR(J1 AND 8)=0)
2580 NY=Y+(K=13 OR(J1 AND 1)=0)-(K=23 OR(J1 AND 2)=0)
2590 IF NX=X AND NY=Y OR NX<1 OR NX>38 OR NY<1 OR NY>23 THEN 2540
2600 POKE P,U:POKE CO+P,UC
2620 X=NX:Y=NY:GOTO 2530
2700 PRINT"[HOME]"
2701 FOR SS=1 TO 22
2702 PRINT" "SPC(38)"  [LEFT]";
2703 NEXT SS
2710 RETURN
2800 PRINT"[CLEAR]"
2810 FOR Y=1 TO 23:FOR X=1 TO 38
2820 IF RND(1)<.2 THEN P=TC+40*Y+X:POKE CO+P,LC:POKE P,81
2830 :NEXT:NEXT
2840 RETURN
3000 GET A$:IF A$=""THEN 3000
3010 RETURN
5000 I=828
5010 READ H$:IF H$="*"THEN 5050
5020 H=ASC(H$)-48:L=ASC(RIGHT$(H$,1))-48
5030 H=H+7*(H>9):L=L+7*(L>9)
5040 POKE I,16*H+L:I=I+1:GOTO 5010
5050 FOR J=0 TO 7:READ X:POKE 675+J,X:NEXT
5060 RETURN
5100 DATA A9,29,85,C3,A5,FB,A9,04,85,C4,A9,D8,85,FC,85,02,A9,00,8D,A1,02,8D,
   A2,02,A8
5110 DATA 20,A3,03,A6,FD,E0,03,F0,08,E0,02,F0,08
5120 DATA A9,00,10,02,A9,00,91,FB,20,CC,03,A0,A2,02,C9,17,D0,E1
5130 DATA A9,29,85,C3,85,FB,A9,04,85,C4,A9,D8,85,FC
5140 DATA 81,FB,29,0F,C9,00,F0,04,A9,20,D0,02,A9,51,D1,C3,F0,02,84,02,91,C3,
   20,CC,03
5150 DATA A5,A2,02,C9,2E,D0,E0,60
5160 DATA 38,A5,C3,E9,29,85,FE,A5,C4,E9,00,85,FF
5170 DATA A2,07,84,FD,BD,A3,02,18,65,FE,85,FE,90,02,E6,FF,B1,FE
5180 DATA C9,51,D0,02,E6,FD,CA,10,E9,60
5190 DATA AE,A1,02,E8,E0,26,D0,14,A2,00,EE,A2,02,18,A5,C3,69,02,90,04,E6,C4,
   E6,FC
5200 DATA 85,C3,85,FB,A6,A1,02,E6,C3,E6,FB,D0,04,E6,C4,E6,FC,60,*
5300 DATA 1,1,38,2,38,1,1,0
```

Hex data pad

Tony Arrowsmith,
Leigh,
Lancashire.

THE AMSTRAD CPC-464 provides the useful facility of re-defining keys to the user's requirements and assigning a string of characters to a given expansion character. Unfortunately, the user manual is so concise that it is difficult to quickly grasp how to go about this procedure.

For example in chapter 8, page 23 we are told how to make "n" print "?". Why on earth would anyone want to do that? Expansion characters 128 to 140 have default locations on the numeric keypad — but how do you find numbers 141 to 152? And what if I want to keep the keypad as it is?

The clue to the answer lies in the diagram on page 16 of Appendix III, where the physical key numbers are listed. This is used as the first parameter in a Key Def statement. Different ASCII or expansion character numbers can be assigned to the normal, shifted or control depressions of each key. The syntax is Key Def key number, repeat (0 or 1), normal, shift, control.

The following short program sets up the numeric keypad as a hexadecimal data entry pad. Existing functions are retained but when shift 1-6 is typed the letters a-f are printed.

(continued on next page)

(continued from previous page)

Shift "." is ",". shift 7, 8 and 9 expand to the keywords Data, Edit and Auto.

The relevent line number should be inserted after Edit and Auto. Control 7, 8 or 9 are all assigned the same function — to escape from a possibly colour-scrambled screen and List in a readable format. Change the ink colours in line 28 to your own favourite combination.

Amstrad hex data pad.

```
10 repeat=0
11 DATA 7,46,44,46
12 DATA 13,49,97,49
13 DATA 14,50,98,50
14 DATA 5,51,99,51
15 DATA 20,52,100,52
16 DATA 12,53,101,53
17 DATA 4,54,102,54
18 DATA 10,55,141,144
19 DATA 11,56,142,144
20 DATA 3,57,143,144
21  FOR k=1 TO 10
22     READ keynumber,normal,shift,control
23     KEY DEF keynumber,repeat,normal,shift,control
24  NEXT k
25 KEY 141,"data "
26 KEY 142,"edit "
27 KEY 143,"auto "
28 KEY 144,"mode 2:border 0:ink 0,0:ink 1,15:list"+CHR$(13)
29 SPEED WRITE 1
30 END:REM **** change to NEW ****
31 SPEED WRITE 1:SAVE "hex pad":SPEED WRITE 0:SAVE "!hex pad"
```

Riski

K Freeman,
Mansfield,
Nottinghamshire.

ORIC

RISKI IS A machine-code game which is run from Basic. It was written on a 48K Oric 1 but should run on the 16K Oric and also the Atmos. The object of the game is to ski down the slope as quickly as possible — use up and down cursor keys to go faster or slower respectively. Then ski through 30 of the approaching gates to complete the course.

Use the cursor keys

Use the right and left cursor keys to control your movement — but be careful, hitting a pole stops the game. The fastest time, elapsed time and gates entered are shown at the top of the screen.

To load the game first type in program 1, which holds the machine code and Pokes it into memory. After entering program 1, run it and if any errors are found you will be informed at which line it lies in — so use the same line numbers, it will help.

If everything is correct then save it and type New but don't switch off the machine. Now enter program 2 and save it. Type

DOKE# 9C,# 265F

Locations # 9C and # 9D hold the end of address of a program for the Oric 1. Now save it with

CSAVE "RISKI",AUTO

It is now ready to run. If you can't be bothered to type it in and would like a copy then send £1.75 to K Freeman, 46 Station Road, Rainworth, Mansfield, Nottinghamshire.

Program 1.

```
1 REM............RISKI...............
3 REM..........program 1.............
10 REM.....Machine Code Loader........
15 PAPER0:INK2:CLS
17 PRINT"Please Wait.....loading data."
100 FORSR=1TO117:READA$,RS
105 FORHX=1TO28STEP2
110 A=VAL("£"+MID$(A$,HX,2))
115 POKE£2000+N,A:N=N+1:CK=CK+A
120 NEXTHX
122 IFRS<>CKTHEN130
125 CK=0:NEXTSR
129 END
130 PRINT"PLEASE CHECK DATA IN LINE......";998+SR*2
132 END
1000 DATA AD1003C9BCF0034C1B20AD0F03C9...=,1351
1002 DATA 7FD0034C2020C9DFD0034C2F2000...=,1420
1004 DATA 004C4C25AD6E21C91FF0F4CE6E21...=,1604
1006 DATA A0074C3820AD6E21C9029F0E5CE6E...=,1638
1008 DATA 21A00E10A9A06D6E218D6C21A9BE...=,1453
1010 DATA 69000D6D21A20480D51219D5221E0...=,1353
1012 DATA 01F004CA4C4D200C5221A2061080...=,1260
1014 DATA 4821692090D4A218D49216900D48...=,1146
1016 DATA 21E000F005CACA4C6020180D6C21...=,1440
1018 DATA 69780D4821AD6D216900D04921AD...=,1311
1020 DATA 6C2180C12080D220AD6D2180C220...=,1540
1022 DATA 8D0320A9048DC420A2000E7C21A0...=,1515
1024 DATA 020D6F21C920D01040B9AFBEC97C...=,1739
1026 DATA F004604CC020604CC32099AFBEC0...=,1765
1028 DATA 55F005E0C84CA9200A4BAE7C21E0...=,1804
1030 DATA 02F01218B07D21A0690480C420CE...=,1515
1032 DATA 7C216BAA4CA920A0AE7021BC5221...=,1445
1034 DATA AE7A21B04821BD21218D0A21BD49...=,1276
1036 DATA 218D22218D0821A200B957214080...=,1154
1038 DATA C78FC920F00FC97C900BA6C920F0...=,1935
1040 DATA 0A10690A4C2021600DC70FE000F0...=,1411
1042 DATA 05C0C04C0521AE7021E004F00CEE...=,1599
1044 DATA 7B21EE7A21EE7A214CEB20A2000E...=,1586
1046 DATA 7B21BE7A216027BF4FBF77BF9FBF...=,1789
1048 DATA C7BF000000000020206720602020...=,757
1050 DATA 206D6E6F7020202020696A6B6C20...=,1060
1052 DATA AFBE0F2061206263642065206620...=,1137
1054 DATA 00000229515SAC1022C004901BC0...=,1003
1056 DATA 0600034CB32110AD4C2269200D4C...=,1142
1058 DATA 22AD4D2269000D4D224CB32138AD...=,1192
1060 DATA 4C22E92080D4C22AD4D22E9000D4D...=,1369
1062 DATA 2218A94E791E22080D521A9226900...=,1105
1064 DATA 0D0621AD4C2200D021AD4D220D09...=,1703
1066 DATA 218E3522A0000958229995800C0C0...=,1668
1068 DATA 0AD0F5CAE000F02518AD05216900...=,1724
1070 DATA 0D0521ADD6216900D0D62110AD00...=,1713
1072 DATA 2169200D0021ADD9216900D0D921...=,1487
1074 DATA 4CD221AC1D22C01AD0002000A2200...=,1348
1076 DATA 004C1922C08C1D2260010A0A0A0A...=,675
1078 DATA 0A00000000000000000000000000...=,10
1080 DATA 00000000000010203040506060606...=,39
1082 DATA 0606060606060606060605040302 01...=,69
1084 DATA 9580202020202020202020207C20...=,814
1086 DATA 20202020202020207C7C2020202020...=,632
1088 DATA 20202020207C7C202020202020207C...=,724
1090 DATA 7C2020202020202020207C7C202020...=,724
1092 DATA 20202020207C30A0B7226D0A226D...=,1168
1094 DATA 8D220D0A622A204B0BA6229D8722CA...=,1725
1096 DATA 10F7291F18C91B000169B28D4C22...=,1522
1098 DATA A9806900000D4D22604040513SCE0...=,1545
1100 DATA 5555555540A4098040A205800C222...=,1442
1102 DATA DDE822F000AFEE2224CEE22A9309...=,1969
1104 DATA E222E000F010CA4CC77223030303030...=,1445
1106 DATA 3030353930353939303720002560A0A...=,801
1108 DATA AA604C03ECA20009309DE22E005...=,1614
1110 DATA F004E04CFD22A2000A0000EE22299...=,1763
1112 DATA C20E00001F00AE005F006E005F000...=,1806
1114 DATA D001CBC0E84C0C2360555554200...=,1477
1116 DATA B0302390000B7C0DFF004C04C2C22...=,1709
1118 DATA 600C1E3F3F1E0C1E3F0101060917...=,435
1120 DATA 2724053F1E3F3F3F1E0000202018...=,488
1122 DATA 2432290020050606C0C1010303020...=,376
1124 DATA 10100C0C06060720202020202020...=,371
1126 DATA 2005050505050505050520140A0502...=,157
1128 DATA 0101000000000020101020050201...=,114
1130 DATA 000000000000201020140A0A0500...=,133
1132 DATA 0102050A141428201020000000000...=,186
1134 DATA 0000000000001020205050A142010...=,101
1136 DATA 2020002C2C2C2C2C2C2C2C050505...=,431
1138 DATA 050505050052C140C050605050404...=,132
1140 DATA 0404042414142C05060504040404...=,164
1142 DATA 040424142C140C0C05040505060 50C...=,195
1144 DATA 14142C2C14240404040404040404...=,216
1146 DATA 0405060605050E142C1424240400...=,205
1148 DATA 00000000000000000404040404040...=,20
1150 DATA 040000000000AD1022C9800034C...=,741
1152 DATA 2B246A0A6E2118690A0A0B7A0EC9...=,1494
1154 DATA 7CF00CEC6E21F004CA4C32244C9B...=,1594
1156 DATA 240A40A2000D5E24106D6E21A0A9...=,1240
1158 DATA 2099A0BEE0003F000E04C4924032A...=,1472
1160 DATA 2BCCA0A0A097C9DA7BE9DCA0EA000...=,1072
1162 DATA AE6E21B990C249DF20EB90F249D1A...=,1014
1164 DATA BFB99224904280C002F00ECDEC04C...=,1920
1166 DATA 7124252425262720292A2BA9FF0D...=,1065
1168 DATA 0004A0AD6E2118690A0A0B7A0EC9...=,1426
1170 DATA 7CF00AEC6E21F004CA4CA2246000...=,1750
1172 DATA 710EC97C00F0CEFA24E5F924A0F9...=,2553
```

Program 2.

```
1 REM..........RISKI.................
2 REM.......program 2.................
10 CLS:POKE618,10:PAPER0:INK2
12 GOSUB500:CLS:PAPER4
18 CALL£2580:CALL£25E9
20 POKE£400,0:CALL£228A:POKE£221D,0
22 CALL£24DE:CALL£22F9:CALL£2528
24 IRQ=DEEK(£FFFE)+1:STI=DEEK(IRQ)
26 DOKE£22F7,STI
40 CALL£2000
50 FORT=1TO4:MUSIC1,2,2,0:PLAY1,0,1,350
52 WAIT50:NEXT
54 MUSIC1,2,6,0:PLAY1,0,1,350
56 WAIT50
98 PLAY0,1,7,30
99 DOKE IRQ,£22C0
100 REPEAT
110 CALL£2180:CALL£2000:CALL£2420
120 CALL£2575
145 PK=PEEK(£400)
150 UNTILPK>0
155 DOKE IRQ,STI
160 IFPK=255THENPLAY0,7,1,350:GOTO250
170 CALL£2500
205 ZAP:PING
250 PLOT10,4,CHR$(6)+"ANOTHER GO Y/N"
255 REPEAT:Q$=KEY$
260 UNTILQ$="Y"ORQ$="N"
265 IFQ$="Y"THEN280
270 END
280 CLS:GOTO18
500 REM.....TITLE PAGE...........
505 PLOT14,3,CHR$(10)+CHR$(3)+"RISKI"
510 PLOT14,4,CHR$(10)+CHR$(6)+"RISKI"
515 PLOT12,6,CHR$(5)+"INSTRUCTIONS"
520 PLOT3,8,"Ski down the slope as fast as you"
525 PLOT3,9,"can passing through 30 gates to "
530 PLOT3,10,"complete the course."
540 PLOT14,12,CHR$(5)+"CONTROL"
545 PLOT3,14,"MOVE RIGHT"+CHR$(3)+"..........CURSOR RIGHT"
550 PLOT3,16,"MOVE LEFT "+CHR$(3)+"..........CURSOR LEFT"
555 PLOT3,18,"FASTER    "+CHR$(3)+"..........CURSOR UP"
560 PLOT3,20,"SLOWER    "+CHR$(3)+"..........CURSOR DOWN"
580 PLOT10,25,"Press"+CHR$(6)+"SPACE"+CHR$(2)+"to play"
590 Q$=KEY$
595 IFQ$<>" "THEN590
600 RETURN
```

```
1174 DATA 24C93AD000EEF024A930BDF924AD...=,1849
1176 DATA FA24C91ED005A97FBD0044CEB24...=,1510
1178 DATA A900BDFA24A930BDF024BDF924AD...=,1837
1180 DATA F024BD01BDADF9240BD2BB603030...=,1945
1182 DATA 0055555555555A0009E222094725...=,1355
1184 DATA 9013F0034C2025994725C005F012...=,1275
1186 DATA C04C0225D9E222994725C005F004...=,1462
1188 DATA C04C1A25A200A000D04725997B0...=,1481
1190 DATA E001F00AE003F006E005F000D001...=,1634
1192 DATA C0C0E04C2C2560303130353439SS...=,1277
1194 DATA C9F7D0034C6625C9BFD017AD0C25...=,1847
1196 DATA C996F010EEBC254C7025AD0C25C9...=,1798
1198 DATA 14F003CE0C25A0004C4B20AEBC25...=,1340
1200 DATA AC0C25B0C000B0FBE000F007CAAC...=,1981
1202 DATA 0C254C7D256064A200BDA1259D18...=,1339
1204 DATA B5E047F004E84C0F25202A236007...=,1420
1206 DATA 060C0C1C1E37312C1613010000000...=,270
1208 DATA 20000000203030303631310900C0C...=,414
1210 DATA 1414163737373F332D1E3F230105...=,520
1212 DATA 051010101412131110222020243F...=,364
1214 DATA 3F0C1A353F1E0C1434220901 1105...=,397
1216 DATA 21A000B90A2699080B9062699 91...=,1437
1218 DATA B0091126999CBDA92099A4DBC005...=,1023
1220 DATA F00FC04CE025474154454 6415354...=,1394
1222 DATA 494D45A9D000D2C2A8D3126A9DDD...=,1514
1224 DATA 7D2A0D322A6A000A201491 7D00B0F...=,1343
1226 DATA A9019D0B0FC00DF01010AD2C2669...=,1550
1228 DATA 2000D2C26D3126A02D26A9000D2D...=,1059
1230 DATA 26BD322AC84C2926A9110DA0BBA9...=,1473
1232 DATA 03BDA9B0A055555555555555555555...=,1361
```

```
   1 REM (PLACE 80 CHARACTERS HE
RE)
  10 PRINT "SIMON COLLINSON PRES
ENTS......"
  20 LET SR=PEEK 16396+256*PEEK
16397+1
  30 POKE SR+165,200
  40 PRINT AT 5,5;"[illegible]"
  50 PRINT AT 6,5;"[illegible]"
  60 PRINT AT 7,5;"[illegible]"
  70 PRINT AT 16,4;"A FIGHT OF S
URVIVAL."
  72 FOR F=1 TO 50
  73 NEXT F
  80 FOR I=1 TO 32
  90 LET A=USR 16537
 100 NEXT I
 110 FOR I=1 TO 5
 120 LET A=USR 16562
 130 NEXT I
 140 PRINT AT 3,0;"A GAME FOR TH
E 16K ZX81 COMPUTER"
 150 POKE 16524,0
 160 POKE 16528,128
 170 LET A=USR 16514
 180 POKE 16524,128
 190 POKE 16528,0
 200 LET A=USR 16514
 202 FOR F=1 TO 50
 203 NEXT F
 210 FOR I=1 TO 20
 220 LET A=USR 16562
 230 NEXT I
 240 POKE 16524,0
 250 POKE 16528,128
 251 CLS
 260 LET A=USR 16514
 270 PRINT AT 11,0;"[illegible]"
 280 LET A$=INKEY$
 290 IF A$="N" THEN GOTO 500
 300 IF A$<>"Y" THEN GOTO 280
 301 PRINT AT 0,10;"[illegible]
"
 310 FOR I=1 TO 22
 320 LET A=USR 16562
 330 NEXT I
 340 PRINT AT 2,0;"USE THE ARROW
 KEYS TO MOVE SID  THE WORM ACRO
SS THE MINE FIELD."
 350 PRINT ,,"[illegible]"
 360 PRINT ,,"THE [illegible] AT
 THE TOP OF  THE SCREEN WILL TE
LL YOU IF ANY MINES ARE NEAR (I.
E. ONE MOVE   AWAY.),SO USE THIS
 WISELY."
 370 PRINT ,,"[illegible]",,"YOU ONLY HAVE
A LIMITED TIME TO REACH THE DOOR
 IN THE MINE-FIELDSO GET A MOVE
ON."
 371 PRINT ,,"[illegible]"
 372 PAUSE 40000
 373 FOR I=1 TO 21
 374 LET A=USR 16562
 375 NEXT I
 380 PRINT AT 2,0;"[illegible]",,"AFTER EAC
H LEVEL THE GAME WILL  GET HARDE
R,UNTIL EVENTUALLY YOU WILL REAC
H SAVETY."
 390 PRINT ,,"[illegible]",,"IF YOU GET ACR
OSS (OR DIE) YOUR ROUTE WILL BE
SHOWN AGAIN,BUT   THIS TIME WITH
 THE MINES VISIBLE...TO SKIP THI
S BIT,JUST PRESS 5"
 400 PRINT ,,"[illegible]",,"YOU CAN BE REI
NCARNATED TWICE   I.E. YOU HAVE
3 LIVES."
 410 PRINT ,,,,"[illegible]"
 420 PAUSE 40000
 430 FOR I=1 TO 22
 440 LET A=USR 16562
 450 LET A=USR 16537
 460 NEXT I
 500 CLS
 501 LET LE=1
 510 LET LI=3
 520 PRINT "[illegible]
     SC=0000"
 530 LET NM=50
 540 LET SC=0
 541 GOSUB 9000
 550 LET AC=15
 551 DIM P(50,2)
 552 LET CO=0
 560 LET UP=21
 561 PRINT AT UP,AC;"S"
 570 LET A$=INKEY$
 571 IF A$<"5" OR A$>"8" THEN GO
TO 570
 580 LET A1=AC
 590 LET U1=UP
 600 LET AC=AC+(A$="8")-(A$="5")
 610 LET UP=UP+(A$="6")-(A$="7")
 611 IF UP=1 AND (AC>14 AND AC<1
8) THEN GOTO 5000
 620 IF UP<>22 THEN GOTO 650
 630 FOR I=1 TO 10
 640 PRINT AT 0,0;"WRONG WAY
   ";AT 0,0;"[illegible]"
 645 NEXT I
 646 PRINT AT 0,0;"[illegible]
"
 647 LET AC=A1
 648 LET UP=U1
 649 GOTO 570
 650 LET SR=PEEK 16396+256*PEEK
16397
 651 IF AC<1 OR AC>30 OR (UP>20
AND AC<>15) OR (UP=1 AND (AC<15
OR AC>17)) THEN GOTO 6500
 652 IF PEEK (SR+(UP*33)+AC+1)=2
53 THEN GOTO 8000
 660 LET MN=0
```

Sid Worm

Simon Paul Collinson,
Mirfield,
West Yorkshire.

ZX-81

THIS PROGRAM is called Sid Worm and is for the 16K ZX-81. As well as it being interesting it is also informative as to the uses of machine-code to enhance games. The Poke on line 30 creates the high-resolution style graphics for the title page.

The idea of the game is to move Sid Worm past the hidden mines. The only way you can tell that there is a mine near, ready to turn you into worm pâté, is by observing the mine-sensor at the top of the screen which will tell you if any mines are one slither away — that's one character. After seven sheets of these you eventually reach the far side of the mine field.

The program works by hiding the mines as character 253, which are Poked onto the screen. The machine code contains a routine which turns these into asterisks.

The machine code is arranged as follows. 16514 changes all the characters on the screen which have the same code as the number in address 16524 into the characters whose address is 16528.

16537 scrolls the screen left, taking the end characters, and putting them at the right. 16562 scrolls the screen down one line, leaving the top line intact, and 16595 makes the mines visible.

```
 670 IF PEEK (SR+((UP+1)*33)+AC+
1)=253 THEN LET MN=1
 680 IF PEEK (SR+((UP-1)*33)+AC+
1)=253 THEN LET MN=MN+1
 690 IF PEEK (SR+(UP*33)+AC+2)=2
53 THEN LET MN=MN+1
 700 IF PEEK (SR+(UP*33)+AC)=253
 THEN LET MN=MN+1
 710 IF MN<>0 THEN PRINT AT 0,0;
CHR$ (MN+156);"[illegible]"
 720 IF MN=0 THEN PRINT AT 0,0;"
[illegible]"
 730 LET CO=CO+1
 740 LET P(CO,1)=UP
 750 LET P(CO,2)=AC
 760 IF CO=50 THEN GOTO 7000
 770 PRINT AT U1,A1;" "
 780 GOTO 561
3000 FOR I=1 TO 21
3010 LET A=USR 16562
3020 NEXT I
3030 CLS
3040 PRINT AT 8,11;"[illegible]";A
T 14,11;"[illegible]"
3050 FOR I=9 TO 13
3060 PRINT AT I,11;"[illegible]";AT I,19;"
[illegible]"
3070 NEXT I
3080 PRINT AT 9,14;"[illegible]";AT 10,15
;"[illegible]";AT 11,14;"[illegible]";AT 12,14;"[illegible]";
AT 13,14;"[illegible]"
3085 PRINT AT 7,11;"SID WORM."
3090 FOR I=1 TO 10
3100 PRINT AT 12,17;"[illegible]"
3110 FOR F=1 TO 5
3120 NEXT F
3130 PRINT AT 12,17;"[illegible]"
3140 FOR F=1 TO 5
3150 NEXT F
3160 NEXT I
3170 PRINT AT 21,0;"WELL DONE..W
ELL DONE..WELL DONE."
3180 FOR I=1 TO 32
3190 LET A=USR 16537
3200 NEXT I
3210 PRINT AT 0,0;"[illegible]"
3220 FOR I=1 TO 22
3230 LET A=USR 16562
3240 NEXT I
3250 LET SC=SC+100
3255 LET NM=NM+200
3260 GOTO 5180
4000 CLS
4010 POKE 16524,0
4020 POKE 16528,128
4030 LET A=USR 16514
4040 PRINT AT 0,12;"[illegible]
"
4045 PRINT AT 1,12;"SCORE=";SC
4050 FOR I=1 TO 50
4060 NEXT I
4070 FOR I=1 TO 21
4080 LET A=USR 16562
4090 NEXT I
4100 CLS
4110 RUN
5000 FOR I=1 TO 32
5010 LET A=USR 16537
5020 NEXT I
5030 PRINT AT UP,AC;"S";AT U1,A1
;" "
5040 LET A=USR 16595
5050 FOR I=1 TO CO
5060 PRINT AT P(I,1),P(I,2);"[illegible]"
5070 IF INKEY$="5" THEN FAST
5080 NEXT I
5090 SLOW
5100 LET SC=SC+(50-CO)
5110 PRINT AT 0,22;"SCORE=";SC
5120 LET NM=NM+100
5130 LET LE=LE+1
5140 IF LE=7 THEN GOTO 3000
5150 FOR I=1 TO 21
5160 LET A=USR 16562
5170 NEXT I
5180 GOSUB 9000
5190 LET AC=15
5200 LET UP=21
5210 GOTO 552
6000 LET A$="WORM"
6010 SAVE A$
6020 RUN
6500 FOR I=1 TO 20
6510 PRINT AT 0,0;"[illegible]..HIT WA
LL";AT 0,0;"[illegible]"
6520 NEXT I
6530 LET AC=A1
6540 LET UP=U1
6545 PRINT AT 0,0;"
   "
6550 GOTO 660
7000 PRINT AT 0,0;"OUT OF TIME"
7001 PRINT AT U1,A1;" "
7010 FOR I=1 TO 20
7020 PRINT AT UP,AC;"[illegible]";AT UP,AC
;"S"
7030 NEXT I
7031 LET LI=LI-1
7032 IF LI=0 THEN GOTO 4000
7040 PRINT AT 0,0;"[illegible]
"
7050 FOR F=1 TO 50
7060 NEXT F
7070 GOSUB 9000
7080 LET UP=21
7090 LET AC=15
7100 GOTO 552
8000 PRINT AT U1,A1;" "
8010 FOR I=1 TO 10
8020 PRINT AT UP,AC;"[illegible]";AT UP,AC
;"S"
8030 NEXT I
8040 PRINT AT UP,AC;"[illegible]"
8050 LET A=USR 16595
8060 FOR I=1 TO CO
8070 PRINT AT P(I,1),P(I,2);"[illegible]"
8080 NEXT I
8085 FOR I=1 TO 10
8090 POKE 16524,0
8100 POKE 16528,128
8110 LET A=USR 16514
8120 POKE 16524,128
8130 POKE 16528,0
8140 LET A=USR 16514
8150 NEXT I
8160 FOR I=1 TO 21
8170 LET A=USR 16562
8180 NEXT I
8190 LET LI=LI-1
8200 IF LI=0 THEN GOTO 4000
8210 GOTO 7040
9000 REM [illegible]
9010 FOR I=1 TO 20
9020 PRINT AT I,31;"[illegible]"
9030 NEXT I
9040 PRINT AT 1,0;"[illegible]";AT 21,0;"[illegible]"
9050 LET SR=PEEK 16396+256*PEEK
16397+1
9051 PRINT AT 0,0;"LAYING MINES
[illegible]"
9060 FOR I=1 TO NM
9061 IF INKEY$="5" THEN FAST
9070 LET P=SR+((INT (RND*17)+3)*
33)+(INT (RND*29)+2)
9090 POKE P,253
9100 NEXT I
9110 SLOW
9120 PRINT AT 2,14;"[illegible]";AT 20
,14;"[illegible]"
9130 FOR I=0 TO 11
9140 PRINT AT 2,13-I;"[illegible]";AT 2,19
+I;"[illegible]";AT 20,13-I;"[illegible]";AT 20,19+I
;"[illegible]"
9150 NEXT I
9160 PRINT AT 2,1;"
  ";AT 20,1;"
  "
9165 PRINT AT 0,0;"[illegible]
  "
9166 PRINT AT 0,21;"SCORE=";SC
9170 RETURN
```

THE FOLLOWING HEX NUMBERS SHOULD BE POKED INTO THE REM ON LINE 1

2A	0C	40	23	0E	16	06	20
7E	FE	00	20	02	36	80	23
10	F6	23	0D	20	F0	C9	2A
0C	40	23	0E	16	7E	23	06
1F	56	2B	72	23	23	10	F9
2B	77	23	23	0D	20	EE	C9
01	06	02	2A	0C	40	09	54
5D	01	B5	02	2A	0C	40	09
ED	B8	2A	0C	40	01	22	00
09	06	20	36	00	23	10	FB
C9	2A	0C	40	23	0E	16	06
20	7E	FE	FD	20	02	36	17
23	10	F6	23	0D	20	F0	C9

Monitor

D P Riley,
Nuneaton,
Warwickshire.

AMSTRAD

THE MONITOR IS unusual in that it is written in Basic, but it provides some of the more fundamental facilities which are handy if you are new to machine code. The program provides the following facilities:

- Query Memory — displays a row of eight bytes of data in hex, starting with the specified address. Hitting Space displays the next eight bytes, and Enter will terminate the command.
- Edit memory — displays the specified address in hex and the data held there, also in hex. Enter will step to the next address without changing any data, and ↑ — unshifted £ — will step backwards to the previous address, again without changing any data. Entering a byte of data, i.e. two hex digits, will cause that value to be entered at the current address. The two digits are entered without a preceding &. Entering a q will terminate the command.
- Breakpoint — sets or clears a breakpoint. If setting, then a four-digit hex address is requested, if clearing, the address is remembered and cleared automatically. On encountering the breakpoint, the contents of all the registers are displayed — except for the alternate register set, which Amstrad discourages the use of.
- Goto — executes a machine-code program from a specified address.
- Load — loads a named binary file from cassette.
- Save — saves a named binary file to cassette.

Except for data entry during the edit command, all values must be entered in hex and preceded by the & character. As programmed, the monitor uses Window # 1 on rows 20 to 25 of the screen. Window # 0 is set to the remainder. Altering lines 25 and 30 will adjust this to your own requirements. The program occupies 2,700 Bytes.

```
10 REM basic machine code monitor
20 REM (c) D.P. Riley 1985
25 WINDOW #0,1,40,1,19
30 WINDOW #1,1,40,20,25
35 FOR pointer=&ABC0 TO &ABDF:READ a:POKE pointer,a:NE
XT
36 DATA &e5,&f5,&e1,&22,&e0,&ab,&e1,&ed
37 DATA &43,&e2,&ab,&ed,&53,&e4,&ab,&22
38 DATA &e6,&ab,&dd,&22,&e8,&ab,&fd,&22
39 DATA &ea,&ab,&ed,&73,&ec,&ab,&e1,&c9
40 CLS #1:LOCATE #1,1,1:PRINT #1,"Command - (q)uery me
mory: (e)dit memory":PRINT #1,"           (b)reakpoint
set: (g)oto":PRINT #1,"
        (l)oad code : (s)ave code
50 INPUT #1,"command ? ",command$
60 IF command$="q" THEN 100 ELSE IF command$="e" THEN
200 ELSE IF command$="b" THEN 300 ELSE IF command$="g"
 THEN 400 ELSE IF comman
d$="l" THEN 500 ELSE IF command$="s" THEN 600
70 GOTO 40
100 REM query memory routine
110 CLS #1:GOSUB 1000:REM get start address
120 byte.count=0:PRINT #1,HEX$(address);"  ";:WHILE by
te.count<8:IF PEEK(address+byte.count)<16 THEN PRINT #
1,"0";
125 PRINT #1,HEX$(PEEK(address+byte.count));" ";:byte.
count=byte.count+1:WEND
130 IF INKEY(18)=-1 AND INKEY(47)=-1 THEN 130 ELSE IF
INKEY(47)=0 THEN PRINT #1:address=address+8:GOTO 120
140 INPUT #1,a$:IF a$="" THEN 40 ELSE 40
200 REM edit memory routine
210 CLS #1:GOSUB 1000
220 PRINT #1,HEX$(address);"  ";HEX$(PEEK(address));"
";:INPUT #1,dat$:IF dat$="q" THEN 40 ELSE IF dat$="" T
HEN address=address+1:GO
TO 220 ELSE 230
230 IF dat$="^" THEN  address=address-1:GOTO 220
240 i=0:WHILE UPPER$(LEFT$(dat$,1))<>HEX$(i) AND i<16:
i=i+1:WEND:IF i>15 THEN 220 ELSE dat=i*16
250 i=0:WHILE UPPER$(RIGHT$(dat$,1))<>HEX$(i) AND i<16
:i=i+1:WEND:IF i>15 THEN 220 ELSE dat=dat+i:POKE addre
ss,dat:address=address+1
:GOTO 220
300 REM breakpoint routine
310 CLS #1:INPUT #1,"(s)et OR (c)lear ",break$:IF brea
k$="s" THEN GOSUB 1000:GOTO 320 ELSE IF break$="c" THE
N 350 ELSE 310
320 break.address=address:saved.data=PEEK(break.addres
s):POKE break.address,&F7:REM set break
330 POKE &30,&C3:POKE &31,&C0:POKE &32,&AB:REM set res
tart 6 ready to jumpto breakpoint handler at &ABC0
340 GOTO 40
350 POKE break.address,saved.data:GOTO 40
400 REM goto routine
410 CLS #1:GOSUB 1000
420 CALL address
430 reg.store=&ABE0:CLS #1:PRINT #1,"A F  B C  D E  H
L  IX   IY   SP   PC"
440 FOR i=reg.store TO reg.store+13 STEP 2:a=PEEK(i+1)
:b=PEEK(i):IF a<16 THEN PRINT #1,"0";
450 PRINT #1,HEX$(a);:IF b<16 THEN PRINT #1,"0";
460 PRINT #1,HEX$(b);" ";:NEXT:PRINT #1,HEX$(break.add
ress);:PRINT #1:LOCATE #1,11,5:PRINT #1,"<Hit any key
when ready>";
470 a$=INKEY$:IF a$="" THEN 470 ELSE 40
500 REM load code routine
510 CLS #1:INPUT #1,"Title ? ",title$
520 WINDOW SWAP 0,1:LOAD title$:WINDOW SWAP 0,1:GOTO 4
0
600 REM save code routine
610 CLS #1:GOSUB 1000:INPUT #1,"length ? ",length:INPU
T #1,"entry point ? ",entry
620 INPUT #1,"title ? ",title$
630 WINDOW SWAP 0,1:SAVE title$,B,address,length,entry
:WINDOW SWAP 0,1
640 GOTO 40
1000 INPUT #1,"Start Address ? ",address
1010 RETURN
9999 LOCATE 1,18
```

Hi-res dump

N Kimberley,
Coal Aston,
Sheffield.

DRAGON

THIS PROGRAM DUMPS the entire hi-res screen to a CGP-115 or MCP-40. It first asks what colour foreground and background you want it copied in. Then it copies it and asks if you want another copy. The copying routine is in machine code and is not relocatable.

I first tried to do a screen dump using Basic but it took some 20 minutes to do one copy. The machine code speeds this up about 10 times.

To enter the machine code first type in the hex loader. Run it and enter the machine code in listing 1. Then save the machine code. Then run it and type in listing 2 and save this after the machine code. To save the machine code type:

CSAVE"M/CODE", 32000,32134,32000

For anyone wanting to know how to output to any printer connected to a Dragon in machine code, load A with the number and JSR $800F.

Hex loader.

```
10 REM HEXLOADER - ENTER THE
20 REM STRING OF HEX DIGITS FIRST
30 REM AND THEN THE CHECKSUM
40 PRINT "ENTER START ADDRESS ";:INPUT START
50 PRINT "ENTER FINISH ADDRESS";:INPUT FINISH
60 FOR N = START TO FINISH STEP 8
65 PRINT N;": ";
70 TT=0:INPUT A$:Z=0
80 FOR G=1 TO LEN(A$) STEP 2
90 P=VAL("&H"+MID$(A$,G,2))
100 TT=TT+P:POKE (N+Z),P
110 Z=Z+1:NEXT
120 PRINT " = ";
130 INPUT T$
150 IF VAL(T$)<>TT THEN PRINT "ERROR - ENTER LINE AGAI
N":GOTO 70
160 NEXT
```

Listing 1.

```
32000 : 8648BD800F860DBD = 874
32008 : 800F8E0600C600A6 = 655
32016 : 80B77D638D0D5CC1 = 974
32024 : 2026F48D238C1E00 = 660
32032 : 26EB39B67D63817F = 992
32040 : 2A208D26B67D6349 = 732
32048 : 877D63B67D648107 = 950
32056 : 270C4CB77D6420E3 = 794
32064 : 108E7D6F8D137F7D = 806
32072 : 6439108E7D658D09 = 691
32080 : 20DA108E7D6A8D01 = 781
32088 : 39A6A0BD800F810D = 857
32096 : 26F73900004A312C = 509
32104 : 300D52312C300D52 = 379
32112 : 2D3235362C2D310D = 353
32120 : 8E0600A68443A780 = 808
32128 : 8C1E0026F63900 = 511
```

```
10 PMODE4,1:PRINT#-2,CHR$(17):PRINT#-2,CHR$(18)
20 CLS:PRINT@9,"SCREEN DUMP"
30 PRINT@41,"====== ===="
40 PRINT:PRINT"  FOR THE DRAGON 32 & CGP115"
50 PRINT"  BY N.KIMBERLEY"
60 PRINT:INPUT"LOAD IN A SCREEN (Y/N)";YN$
70 IF YN$="N" THEN 90
80 SCREEN1,1:PCLS:CLOADM
90 CLS:PRINT:PRINT"ENTER FORGROUND COLOUR":GOSUB230:IN
PUTFC
100 PRINT:PRINT"ENTER BACKGROUND COLOUR":GOSUB230:INPU
TBC
110 IF FC=BC OR FC=4 AND BC=4 THEN PRINT"INVALID INPUT
":SOUND1,6:GOTO90
120 SCREEN1,1
130 IF FC=4 THEN GOTO160
140 PRINT#-2,"C";FC
150 EXEC32000
160 IF BC=4 THEN 190
170 PRINT#-2,"C";BC
180 EXEC32120:EXEC32000
190 EXEC32120:CLS:INPUT"ANOTHER COPY (Y/N)";YN$
200 IF YN$="N" THEN CLS:END
210 PRINT#-2,"A":PRINT#-2,STRING$(5,11):PRINT#-2,CHR$(
18)
220 GOTO130
230 PRINT"0 BLACK","1 BLUE","2 GREEN","3 RED","4 PRINT
ER PAPER COLOUR"
240 RETURN
```

Image Edit

C Denning,
Wilmslow,
Cheshire.

SPECTRUM

HAVE YOU EVER wished you could combine the individual pictures you have created and stored on cassette? This is difficult in Basic as you cannot merge two screens or load one with Over on. This program, Image Edit for the 48K Spectrum lets you combine pictures.

First type in program 1. Save it using the command

SAVE "Image Edit" LINE 9000

Then Verify it. Next, to enter the machine code, type in and Run the hex loader. Type in the hex code and save with

SAVE "Image Edit" CODE 23296,188

Now Verify the code.

Once both parts of the program have been saved, enter

RANDOMIZE USR 0

to reset the computer. Then rewind the cassette and load the complete program.

You should be shown a menu. If you want to combine 2, 3 or 4 images, load them into different screens. To combine more than four images, load them in stages of four, Saving the composite on cassette each time. Later, combine the composite images.

When you have loaded all the images you want, press S to start the machine-code program. There are two different ways of combining images, using the machine code commands OR or XOR.

XOR puts the images together in the same way as the Basic command Over. Where there is an overlap, a space is made. OR adds the images completely, leaving the overlap black. When using the machine-code program, press 1, 2, 3 or 4 to put an image on the screen. It will be added to what is already on the screen according to the mode — XOR or OR.

- 8 and 9 put the program in XOR and OR modes respectively.
- 6 returns to the Basic program.
- 0 clears the screen.

When returning to the Basic program, you will be given the opportunity of Saving the composite or Loading more images.

It is possible to create some other effects by combinations of the two modes. For example, you can subtract an image from what is already on the screen adding it first in OR mode, then in XOR mode.

When combining the pictures, colours are ignored as they would be jumbled when combined. The final image is Saved as black on white.

Hex loader.

```
  10 DEF FN h(h$)=16*(CODE h$(1)
-48-(7 AND h$(1)>"9"))+CODE h$(2
)-48-(7 AND h$(2)>"9")
  20 INPUT "Start ";s
  30 INPUT "Finish ";f
  40 FOR n=s TO f STEP 8
  50 LET tot=0: PRINT n;": ";
  60 INPUT h$: PRINT h$;
  70 LET x=0
  80 FOR b=1 TO LEN h$ STEP 2
  90 LET z=FN h(h$): LET tot=tot
+z
 100 POKE n+x,z
 110 LET h$=h$(3 TO ): LET x=x+1
 120 NEXT b
 130 PRINT " = ";: INPUT t: PRIN
T t
 140 IF tot<>t THEN PRINT "input
 error - try again": GO TO 50
 150 NEXT n
 160 REM enter STOP to stop
```

Program 1.

```
  10 POKE 23658,8: CLS : PRINT A
T 5,14;"Menu":'''"1/2/3/4    Load
 Screen"''"S           Start Edito
r"''"0            Quit"
  20 IF INKEY$="0" THEN STOP
  30 IF INKEY$="S" THEN GO TO 10
0
  40 LET C=CODE INKEY$: IF C>48
AND C<53 THEN LET sc=(CODE INKEY
$)-49: PRINT #0;"Start tape.": L
OAD ""CODE sc*6144+40000: GO TO
10
  50 GO TO 20
 100 RANDOMIZE USR 23296: PRINT
#0;"Save Image Y/N ? "
 110 IF INKEY$="Y" THEN INPUT "N
ame? ";n$: SAVE n$SCREEN$ : GO T
O 10
 120 IF INKEY$="N" THEN GO TO 10
 130 GO TO 110
9000 BORDER 7: PAPER 7: INK 0: C
LEAR 39999: PRINT #0;"Leave tape
 running.": LOAD ""CODE 23296: G
O TO 10
```

The machine code.

```
23296: 06F70EFE78ED78E6 = 1228
23304: 01283578ED78E602 = 803
23312: 284578ED78E60428 = 860
23320: 5578ED78E6082866 = 943
23328: 06EF78ED78E610C8 = 1168
23336: 78ED78E6042867D78 = 996
23344: ED78E602287E78ED = 1112
23352: 78E601CC6B0D18C0 = 891
23360: 1100402140 9C0100 = 335
23368: 181ACD9E5B121323 = 576
23376: 0B78B120F418A911 = 794
23384: 0040214064010018 = 366
23392: 1ACD9E5B1213230B = 563
23400: 78B120F418921100 = 760
23408: 402140CC0100181A = 416
23416: CD9E5B1213230B78 = 657
23424: B120F4C3005B1100 = 756
23432: 402140E40100181A = 440
23440: CD9E5B1213230B78 = 657
23448: B120F4C3005BF53A = 1042
23456: 815CCB472803F1B6 = 961
23464: C9F1AEC93E003281 = 1058
23472: 5CC3005E3E013281 = 620
23480: 5CC3005B5B = 469
```

Turtles

Ian S Gibson,
Maidstone,
Kent.

ELECTRON

THIS PROGRAM ENABLES the BBC B or Electron to perform several features associated with Turtle graphics. When you run the program, the micro's name appears with a prompt. Electron users should change Mode 7 statement to Mode 6 when typing in. The following commands are available in memory from the start. They can be used to build up other user defined words. These resident words are as follows:

Word	Abbreviation	Meaning
Circle	C	Draws a circle, centre graphics cursor, radius R. F is 0 for unfilled, 1 for filled.
Fill	—	Fill area of screen above and below graphics cursor.
Left X	L	Rotate cursor X° left — anti-clockwise.
Right X	R	Rotate X° right — clockwise.
Forward X	F	Draw line forward X units.
Pencil X	P	Change colour of graphics, 0-3.
Clean	—	Clear the screen.
Save X$	—	Save all user-defined words under the name, X$.
Load X$	—	Load program marked X$.
New	—	Erases all user-defined words.
Move X	—	Move cursor forward X units without drawing.
Plot	—	Plot cursor position.
Home	—	Return cursor to bottom left-hand corner
Colour XY	—	Change logical colour X to actual colour Y.
Wait X	—	Pause for X units.
Triangle X	T	Make a triangle; the third point is X units ahead, and the other two are the last two points plotted or moved to.
Scale X	—	Changes the scale of future drawing — 1 is default.
Dot	—	Makes future lines dotted.

(continued on next page)

(continued from previous page)

Line	—	Makes future lines undotted.
Forget X$	—	Forgets the user defined word, X$.
Text X$	—	Writes text at graphics cursor.

The following commands may not be used in user-defined words.

List	—	Lists all user-defined words.
Goodbye	Bye	Returns to Basic.
Build X$	B	Creates or edits a user-defined word with the name X$.

Save completed picture

To save a completed picture without the user defined words, type Bye and Return followed by

*SAVE"name"5800 7FFF

and reloading by

*Load "Name".

To create a user-defined word, X$, the following procedure should be used.

BUILD X$ — where X$ is a name
Select a line number 1-20.

The contents of that line may be any user-defined, or resident word in Turtle's dictionary, for example F 100. In addition, after selecting a line number, Insert F0 or Delete F9 are available. Insert moves all lines after and including the selected line up one place. Line 20 is lost. Delete deletes the selected line thus moving all subsequent lines down one place. Return exits the editor when used instead of a line number.

Here is an example of a user-defined word:

BUILD STEP
1 (RETURN)
FORWARD 50 (RETURN)
2 (RETURN)
RIGHT 90 (RETURN)
3 (RETURN)
FORWARD 50 (RETURN)
4 (RETURN)
LEFT 90 (RETURN) (RETURN)

Type
HOME (RETURN)
STEP (RETURN)
should produce a step.

User-defined words repeated

User-defined words can be repeated by putting a number after them. For example Step 5 and Return produces five more steps. Once a word has been defined, it may be used in any other user-defined word. For example,

BUILD STEPS (RETURN)
1 (RETURN)
HOME (RETURN)
2 (RETURN)
STEP 5 (RETURN) (RETURN)
STEPS (RETURN)
produces five steps.

Complex patterns can be built up in this way very easily. Programs may be merged simply by loading and running two sets of saved words to produce a final picture.

```
   10REM TURTLE GRAPHICS
   20REM For the BBC B or Electron
   30REM By Ian S. Gibson
   40*KEY10OLD|MRUN|M
   50ON ERROR GOTO 1610
   60MODE 7
   70H%=HIMEM
   80MODE 5
   90VDU 28,0,31,19,27
  100PRINT
  110IF H%=31744 THEN PRINT"BBC Microcomputer" ELSE PR
INT"Acorn Electron"
  120PRINT
  130PRINT"I.S.G. TURTLE"
  140PRINT
  150REM Initialise variables
  160*KEY0"f0|M"
  170*KEY9"f9|M"
  180DIM D$(100,20),N$(100)
  190X%=0:Y%=0:A%=0:M%=0:Z=1
  200VDU 28,0,31,19,27
  210VDU 24,0;160;1279;1023;
  220VDU 29,0;160;
  230INPUT LINE">"I$
  240IF I$="" THEN 230
  250IF I$="GOODBYE" OR I$="BYE" THEN 1170
  260PROCdo(I$):IF T%=TRUE THEN 230
  270IF H$="BUILD" OR L$="B " THEN MODE 7:PROCbu:MODE
5:GOTO 200
  280IF H$="LIST" THEN MODE 7:PROCli:MODE 5:GOTO 200
  290PROCtr(I$)
  300GOTO 230
  310DEF PROCtr(I$)
  320LOCAL I%,J%,K%,L%,D%,E%,E$
  330L%=INSTR(I$," ")
  340IF L%<>0 THEN E%=VAL(MID$(I$,L%+1)):I$=LEFT$(I$,L
%-1) ELSE E%=1
  350D%=0
  360D%=D%+1
  370I%=0
  380I%=I%+1
  390IF N$(I%)=I$ THEN 420
  400IF I%<100 AND N$(I%)<>"" THEN 380
  410PRINT"The word ";I$''"doesn't exist":PROCerr:ENDPR
OC
  420J%=0
  430J%=J%+1
  440E$=D$(I%,J%)
  450IF E$="" THEN 480
  460PROCdo(E$)
  470IF T%=FALSE THEN PROCtr(E$)
  480IF J%<>20 THEN 430
  490IF D%<>E% THEN 360
  500ENDPROC
  510DEF PROCdo(A$)
  520T%=FALSE
  530L$=LEFT$(A$,2)
  540S%=INSTR(A$," ")
  550R$=MID$(A$,S%+1)
  560R=VAL(R$)
  570R%=VAL(R$)
  580H$=LEFT$(A$,S%-1)
  590S1%=INSTR(R$," ",2)
  600R1$=MID$(R$,2+LEN(STR$(R%)))
  610R1%=VAL(R1$)
  620IF H$="CIRCLE" OR L$="C " THEN PROCcircle(X%,Y%,R
%*Z,R1%):T%=TRUE:ENDPROC
  630IF H$="FILL" THEN PROCfill(X%,Y%):T%=TRUE:ENDPROC
  640IF L$="L " OR H$="LEFT" THEN PROCt(-R%):T%=TRUE:E
NDPROC
  650IF L$="R " OR H$="RIGHT" THEN PROCt(R%):T%=TRUE:E
NDPROC
  660IF L$="F " OR H$="FORWARD" THEN PROCf(R%):T%=TRUE
:ENDPROC
  670IF L$="P " OR H$="PENCIL" THEN GCOL 0,R%:T%=TRUE:
ENDPROC
  680IF H$="CLEAN" THEN CLG:CLS:T%=TRUE:ENDPROC
  690IF H$="FORGET" THEN PROCft(R$):T%=TRUE:ENDPROC
  700IF H$="SAVE" THEN PROCsa(R$):T%=TRUE:ENDPROC
  710IF H$="LOAD" THEN PROClo(R$):T%=TRUE:ENDPROC
  720IF H$="NEW" THEN PROCne:T%=TRUE:ENDPROC
  730IF H$="MOVE" OR L$="M " THEN PROCmo(R%):T%=TRUE:E
NDPROC
  740IF H$="PLOT" THEN PROCpl(R%):T%=TRUE:ENDPROC
  750IF H$="HOME" THEN X%=0:Y%=0:A%=0:MOVE 0,0:T%=TRUE
:ENDPROC
  760IF H$="COLOUR" THEN VDU 19,R%,R1%,0,0,0:T%=TRUE:E
NDPROC
  770IF H$="TEXT" THEN PROCte(R$):T%=TRUE:ENDPROC
  780IF H$="WAIT" THEN FOR P=1 TO R%:NEXT P:T%=TRUE:EN
DPROC
  790IF H$="TRIANGLE" OR L$="T " THEN PROCtri(R%):T%=T
RUE:ENDPROC
  800IF H$="SCALE" THEN Z=R:T%=TRUE:ENDPROC
  810IF H$="DOT" THEN M%=1:T%=TRUE:ENDPROC
  820IF H$="LINE" THEN M%=0:T%=TRUE:ENDPROC
  830ENDPROC
  840DEF PROCt(F%)
  850LET A%=A%+F%
  860IF A%<0 THEN REPEAT A%=360-A%:UNTIL A%<360
  870A%=A% MOD 360
  880ENDPROC
  890DEF PROCmo(F%)
  900PROCha
  910IF X1%<0 OR X1%>1279 OR Y1%<0 OR Y1%>863 THEN PRI
NT "Can't - moves off"'"edge of screen":PROCerr
  920X%=X1%
  930Y%=Y1%
  940MOVE X%,Y%
  950ENDPROC
  960DEF PROCf(F%)
  970PROCha
  980IF X1%<0 OR X1%>1279 OR Y1%<0 OR Y1%>863 THEN PRI
NT "Can't - line off"'"edge of screen":PROCerr:ENDPROC
  990X%=X1%
 1000Y%=Y1%
 1010IF M%=1 THEN PLOT 29,X%,Y% ELSE DRAW X%,Y%
 1020ENDPROC
 1030DEF PROCpl(F%)
 1040PROCha
 1050IF X1%<0 OR X1%>1279 OR Y1%<0 OR Y1%>863 THEN PRI
NT "Can't - plot off"'"edge of screen":PROCerr:ENDPROC
 1060X%=X1%
 1070Y%=Y1%
 1080PLOT 69,X%,Y%
 1090ENDPROC
 1100DEF PROCtri(F%)
 1110PROCha
 1120IF X1%<0 OR X1%>1279 OR Y1%<0 OR Y1%>863 THEN PRI
NT "Can't - triangle"'"off edge of screen":ENDPROC
 1130X%=X1%
 1140Y%=Y1%
 1150PLOT 85,X%,Y%
 1160ENDPROC
 1170PRINT
 1180PRINT"BASIC"
 1190PRINT
 1200GOTO 2070
 1210DEF PROCbu
 1220IF R$="BUILD" THEN REPEAT:INPUT"Build what";R$:UN
TIL R$<>"" AND R$<>"BUILD":CLS
 1230PRINT"BUILD ";R$
 1240O%=0
 1250O%=O%+1
 1260IF N$(O%)=R$ THEN 1290
 1270IF O%<100 AND N$(O%)<>"" THEN 1250
 1280IF O%=100 AND N$(O%)<>"" THEN PRINT"There is no m
ore room to store any words.":ENDPROC
 1290FOR F%=1 TO 20
 1300PRINTTAB(0,F%+1);F%;" : ";D$(O%,F%)
 1310NEXT
 1320PRINTTAB(0,23);SPC(39):INPUTTAB(0,23);"Enter line
 number or press RETURN :"T$
 1330IF T$="" THEN N$(O%)=R$:ENDPROC
 1340T%=VAL(T$)
 1350IF T%<1 OR T%>20 THEN 1320
 1360PRINTTAB(0,24);"f0 - INSERT","f9 - DELETE";TAB(0,
T%+1);SPC(39);TAB(0,T%+1);T%;" : ";
 1370INPUT LINE""U$
 1380PRINTTAB(0,24);SPC(38);
 1390IF U$="f0" THEN PROCin:GOTO1320
 1400IF U$="f9" THEN PROCde:GOTO1320
 1410D$(O%,T%)=U$
 1420GOTO 1320
 1430DEF PROCft(U$)
 1440I%=0
 1450I%=I%+1
 1460IF N$(I%)=U$ THEN 1490
 1470IF I%<100 AND N$(I%)<>"" THEN 1450
 1480PRINT"The word ";U$''"doesn't exist":PROCerr:ENDPR
OC
 1490IF I%=100 THEN 1560
 1500FOR R%=I% TO 99
 1510N$(R%)=N$(R%+1)
 1520FOR S%=1 TO 20
 1530D$(R%,S%)=D$(R%+1,S%)
 1540NEXT
 1550NEXT
 1560N$(100)=""
 1570FOR S%=1 TO 20
 1580D$(100,S%)=""
 1590NEXT
 1600ENDPROC
 1610IF ERR<>17 THEN 1650
 1620IF HIMEM<>22528 THEN MODE 5:PRINTTAB(0,30);
 1630PRINT'"ESCAPE"
 1640GOTO200
 1650MODE 7
 1660PRINT
 1670REPORT:PRINT" at line ";ERL
 1680GOTO2070
 1690DEF PROCsa(U$)
 1700IF U$="SAVE" THEN REPEAT:INPUT"Save what";U$:UNTI
L U$<>"" AND U$<>"SAVE"
 1710Y%=OPENOUT(U$)
 1720PRINT'"Saving ";U$
 1730PRINT#Y%,U$
 1740FOR I%=1 TO 100
 1750IF N$(I%)="" THEN 1800
 1760PRINT#Y%,N$(I%)
 1770FOR G%=1 TO 20
 1780PRINT#Y%,D$(I%,G%)
 1790NEXT
 1800NEXT
 1810CLOSE#Y%
 1820ENDPROC
 1830DEF PROClo(U$)
 1840PRINT'"Searching"
 1850IF U$="LOAD" THEN U$=""
 1860Y%=OPENIN(U$)
 1870INPUT#Y%,U$
 1880PRINT'"Loading ";U$
 1890I%=0
 1900IF EOF#Y% THEN 1970
 1910I%=I%+1
 1920INPUT#Y%,N$(I%)
 1930FOR T%=1 TO 20
 1940INPUT#Y%,D$(I%,T%)
 1950NEXT
 1960GOTO 1900
 1970CLOSE#Y%
 1980ENDPROC
 1990DEF PROCne
 2000PRINT"NEW - Are you sure?";
 2010REPEAT
 2020G$=GET$
 2030UNTIL G$="Y" OR G$="N"
 2040PRINTG$
 2050IF G$="N" THEN ENDPROC
 2060RUN
 2070DEF PROCli
 2080PRINT"LIST"''"Press SHIFT to scroll if necessary"'
 2090VDU14
 2100FOR I%=1 TO 100
 2110IF N$(I%)<>"" THEN PRINTN$(I%)
 2120NEXT
 2130VDU15
 2140PRINT''"Press RETURN  ";
 2150REPEAT UNTIL GET=13
 2160ENDPROC
 2170DEF PROCin
 2180IF T%=20 THEN D$(O%,20)="":PRINTTAB(0,21);SPC(39)
;TAB(0,21);"20 :":ENDPROC
 2190FOR U%=20 TO T%+1 STEP-1
 2200D$(O%,U%)=D$(O%,U%-1)
 2210PRINTTAB(0,U%+1);SPC(39);TAB(0,U%+1);U%;" : ";D$(
O%,U%)
 2220NEXT
 2230D$(O%,T%)=""
 2240PRINTTAB(0,T%+1);SPC(39);TAB(0,T%+1);T%;" : ";D$(
O%,T%)
 2250ENDPROC
 2260DEF PROCde
 2270IF T%=20 THEN PRINTTAB(0,21);SPC(39);TAB(0,21);"2
0 :":D$(O%,20)="":ENDPROC
 2280FOR U%=T% TO 19
 2290D$(O%,U%)=D$(O%,U%+1)
 2300PRINTTAB(0,U%+1);SPC(39);TAB(0,U%+1);U%;" : ";D$(
O%,U%)
 2310NEXT
 2320T%=20:GOTO 2270
 2330DEF PROCte(F$)
 2340VDU5
 2350PRINTF$
 2360VDU4
 2370MOVE X%,Y%
 2380ENDPROC
 2390DEF PROCha
 2400X1%=X%+Z*(F%*(SIN(RAD(A%))))
 2410Y1%=Y%+Z*(F%*(COS(RAD(A%))))
 2420ENDPROC
 2430DEF PROCerr
 2440PRINT"RETURN to continue ";
 2450REPEAT UNTIL GET=13
 2460PRINT'
 2470ENDPROC
 2480DEF PROCcircle(X%,Y%,O%,W%)
 2490LOCAL I%,J%,OX1%,OX2%
 2500OX1%=X%:OX2%=X%
 2510MOVE X%,Y%+O%
 2520FOR I%=Y%+O%-4 TO Y%-O% STEP-4
 2530J%=SQR(ABS(O%*O%-(I%-Y%)*(I%-Y%)))
 2540IF W%=0 THEN PLOT 69,OX1%,I%+4
 2550OX1%=X%-J%
 2560PLOT 5,OX1%,I%
 2570IF W%=0 THEN PLOT 69,OX2%,I%+4
 2580OX2%=X%+J%
 2590PLOT 5,OX2%,I%
 2600NEXT
```

```
2610MOVE X%,Y%
2620ENDPROC
2630DEF PROCfill(X%,Y%)
2640LOCAL I%,J%,K%,P%
2650I%=Y%:J%=X%:P%=POINT(X%,Y%)
2660REPEAT
2670GOSUB2760
2680I%=I%+4
2690UNTIL I%>1023 OR POINT(J%,I%)<>P%
2700I%=Y%
2710REPEAT
2720GOSUB2760
2730I%=I%-4
2740UNTIL I%<0 OR POINT(J%,I%)<>P%
2750ENDPROC
2760REPEAT
2770J%=J%-8
2780UNTIL POINT(J%,I%)<>P% OR J%<=0
2790PLOT 69,J%+8,I%
2800J%=X%
2810REPEAT
2820J%=J%+8
2830UNTIL POINT(J%,I%)<>P% OR J%>=1279
2840DRAW J%-8,I%
2850J%=X%
2860RETURN
2870END
```

Microdrive catalogue

Per Arne Jenseen,
Aalesund, Norway.

THE PROGRAM for the ZX 16/48K Spectrum generates a list of programs on a Microdrive cartridge together with the usual information from the Cat command. It also helps you to Load and Erase programs on your cartridge plus some more functions.

Saved on Microdrive cartridge

The program should be Saved on a cartridge like this:

SAVE *"m";1;"run" LINE 1:OPEN # 4;"m";1;CHR$ 0 + "cat":CAT # 4;1:CLOSE # 4

Type New and then Run. The program will then autoload and give you the opportunity to use your Microdrive with a minimum of effort. You will be presented with a list of programs on the cartridge. Pressing the symbol in front of each program name will automatically load it. If the name ends with a full stop, the program will believe it's a block of code, and will ask you for a clear address. This can be skipped by pressing the enter key. The program will then not New itself thus preserving the code block.

If you want to delete a program press delete and the program symbol on the list. The program will then be erased.

You can operate more than one Microdrive by pressing Enter. A prompt will appear at the bottom of the screen to tell you when you can do this. Caps+6 will bring you back to drive 1.

To add a program to the cartridge, Save the program as normal. Then type New and Run. Press Edit and the new program will be handled by the program. You can also use the Edit after doing a normal erase from Basic as the program will believe the other program is still on the cartridge.

You can scroll the list

If the cartridge is very full and the list is long, then you can scroll the list by using the enter and Caps+7.

The program must be held in Microdrive 1. To use other drives you must create a Cat file on the other cartridge. This is done by the same procedure described, but cancelling the Saving. Use drive 1 when doing this.

If you do this you can have my program on one cartridge only and Cat files on all the others.

```
   1 REM xxxxxxxxxxxxxxxxxxxxxxxxxx
   2 REM RUN PROGRAM TO
         ZX MICRODRIVE
   5 POKE 23658,8
  10 DIM c$(54,10): LET md=1
  20 LET an=1: PRINT AT 21,14; F
LASH 1;"CAT ";md
  22 OPEN #4;"m";md;CHR$ 0+"cat
"
  30 INPUT #4;c$(an)
  35 IF an>2 AND c$(an,1)=" " TH
EN LET an=an+1: INPUT #4;c$(an):
 CLOSE #4: GO TO 50
  40 LET an=an+1: GO TO 30
  50 REM PRINT CAT
  55 CLS #
  60 GO SUB 1600
  70 FOR n=3 TO an-2
  80 IF n<13 THEN PRINT n-3;"=";
c$(n),: GO TO 110
  90 PRINT CHR$ (n+52);"=";c$(n)
,
 100 IF SCREEN$ (20,16)<>" " AND
 n<an-2 THEN PRINT #0; INVERSE 1
;AT 0,3;"PRESS ENTER FOR NEXT SC
REEN": GO SUB 113
 110 NEXT n
 111 PRINT #0; INVERSE 1;"PRESS
ENTER FOR NEXT MICRODRIVE"
 112 REM INKEY$ TEST
 113 LET de=0
 115 PAUSE 0: LET i$=INKEY$: IF
i$<>CHR$ 12 THEN GO SUB 1500
 120 IF i$=CHR$ 13 AND n>=an-2 A
ND md<8 THEN LET md=md+1: GO TO
20
 125 IF i$=CHR$ 13 AND n>=an-2 T
HEN RUN
 130 IF i$=CHR$ 13 THEN CLS : GO
 SUB 1600: RETURN
 140 IF i$=CHR$ 7 THEN PRINT AT
21,14; FLASH 1;"EDIT": GO TO 100
0
 150 IF i$=CHR$ 10 THEN RUN
 160 IF i$=CHR$ 11 THEN GO TO 50
 165 IF i$=CHR$ 12 THEN PRINT AT
 21,2; FLASH 1;"WHAT DO YOU WANT
 TO ERASE?": LET de=1: GO TO 115
 170 IF CODE i$<48 OR CODE i$>an
+50 THEN GO TO 115
 180 IF CODE i$>57 AND CODE i$<6
5 THEN GO TO 113
 200 REM NAMEVARIABEL'S
 210 IF CODE i$<=57 THEN LET lo=
3+VAL i$: GO TO 230
 220 LET lo=CODE i$-52
 230 LET b$=c$(lo)
 240 FOR n=10 TO 1 STEP -1
 250 IF b$(n)=" " THEN NEXT n
 260 LET b$=b$( TO n)
 261 IF de=1 THEN PRINT FLASH 1;
AT 21,9;"ERASE ";b$: ERASE "m";m
d;b$: GO TO 1000
 262 REM LOADING
 263 POKE 23658,0
 265 PRINT FLASH 1;AT 21,7;"LOAD
ING ";b$
 270 IF b$(n)="." THEN GO TO 290
 280 LOAD *"m";md;b$
 290 INPUT "CLEAR ", LINE q$
 300 IF CODE q$>=50 AND CODE q$<
=54 AND LEN q$=5 THEN GO TO 320
 310 LOAD *"m";md;b$CODE : STOP
 320 LOAD *"m";md;b$CODE VAL q$+
1: CLEAR VAL q$: PAUSE 20: NEW
1000 REM EDIT
1010 ERASE "m";md;CHR$ 0+"cat"
1020 OPEN #4;"m";md;CHR$ 0+"cat"
1030 CAT #4;md
1040 CLOSE #4
1045 GO SUB 1530
1050 GO TO 20
1500 REM DELETE ?
1505 LET rem=PEEK 23635+256*PEEK
 23636+5: LET xx=USR rem
1530 PRINT AT 21,0;"
                         "
1540 RETURN
1600 REM HEADLINE
1610 LET le=3
1630 IF c$(an,le)=" " THEN LET l
e=le-1: GO TO 1630
1640 LET len=10
1650 IF c$(1,len)=" " THEN LET l
en=len-1: GO TO 1650
1660 PRINT "MICRODRIVE ";md;": ";
c$(1 TO len);" ("; FLASH 1;VAL c
$(an, TO le)-1;"K Left"; FLASH 0
;")"
"???": REM The ?'s is Graphic In
verse 3
1670 RETURN
9997 REM FIRST RUN LINE 9998 THE
N DELETE LINE 9997-9999
9998 RESTORE : LET rem=PEEK 2363
5+256*PEEK 23636+5: FOR N=rem TO
 rem+11: READ A: POKE N,A: NEXT
N
9999 DATA 33,224,90,62,63,6,31,1
19,35,16,252,201
```

Symbol Print

M J Bennett,
Jeddah 21452,
Saudi Arabia.

DECIPHERING the Commodore 64 graphics/control symbols printed in listings is often very difficult. This program is designed to make them more readable.

It has been written for the Epson RX-80, but would run on any standard printer with very few alterations.

The program is designed to read only from disc, but it would be easy to transfer taped programs to disc for this purpose.

```
20 POKE 53280,0:POKE 53281,0:PRINT"■"
100 DATA 19,147,17,145,29,157,18,146,20,148,141,32
110 REM 80-COLUMN CURSOR STUFF
120 DATA 129,149,150,151,152,14,142,153,154
130 DATA 155
140 REM VIC STUFF
150 DATA 144,5,28,159,156,30,31,158
160 DATA 8,9,133,137,134,138,135,139,136,140
161 DATA 92,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110
162 DATA 111,112,113,114,115,116,117,118,119,120,121,122,123,124,125,126,127
163 DATA 161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177
164 DATA 178,179,180,181,182,183,184,185,186,187,188,189,190,191
170 DATA HOME,CLEAR,DOWN,UP,RIGHT,LEFT,RVS,RVOFF,DEL,INST,RETURN,SPACE
180 DATA ORANGE,BROWN,L.RED,GREY1,GREY2,TEXT,GRAPHIC,L.GREEN,L.BLUE
190 DATA GREY3
200 DATA BLACK,WHITE,RED,CYAN,MAGENTA,GREEN,BLUE,YELLOW
210 DATA LOCK,UNLOCK,F1,F2,F3,F4,F5,F6,F7,F8
211 DATA POUND,LEFT ARROW,SHIFT/*,SHIFT/A,SHIFT/B,SHIFT/C,SHIFT/D,SHIFT/E
212 DATA SHIFT/F,SHIFT/G,SHIFT/H,SHIFT/I,SHIFT/J,SHIFT/K,SHIFT/L,SHIFT/M
213 DATA SHIFT/N,SHIFT/O,SHIFT/P,SHIFT/Q,SHIFT/R,SHIFT/S,SHIFT/T,SHIFT/U
214 DATA SHIFT/V,SHIFT/W,SHIFT/X,SHIFT/Y,SHIFT/Z,SHIFT/+,COM/-,SHIFT/-
215 DATA SHIFT/↑,COM/*,COM/K,COM/I,COM/T,COM/@,COM/G,COM/+,COM/M
216 DATA COM/POUND,SHIFT/POUND,COM/N,COM/Q,COM/D,COM/Z,COM/S,COM/P,COM/A
217 DATA COM/E,COM/R,COM/W,COM/H,COM/J,COM/L,COM/Y,COM/U,COM/O,SHIFT/@
218 DATA COM/F,COM/C,COM/X,COM/V,COM/B
220 DIM A(105),A$(105),K$(91)
230 FOR J=0 TO 104:READ A(J):NEXT
240 FOR J=0 TO 104:READ A$(J):NEXT
250 DATA END,FOR,NEXT,DATA,INPUT#,INPUT,DIM,READ,LET,GOTO,RUN,IF,RESTORE,GOSUB
260 DATA RETURN,REM,STOP,ON,WAIT,LOAD,SAVE,VERIFY,DEF,POKE,PRINT#,PRINT,CONT
270 DATA LIST,CLR,CMD,SYS,OPEN,CLOSE,GET,NEW,TAB(,TO,FN,SPC(,THEN,NOT,STEP
280 DATA +,-,*,/,↑,AND,OR,>,=,<,SGN,INT,ABS,USR,FRE,POS,SQR,RND,LOG,EXP,COS
290 DATA SIN,TAN,ATN,PEEK,LEN,STR$,VAL,ASC,CHR$,LEFT$,RIGHT$,MID$,GO,CONCAT
300 DATA DOPEN,DCLOSE,RECORD,HEADER,COLLECT,BACKUP,COPY,APPEND,DSAVE,DLOAD
310 DATA CATALOG,RENAME,SCRATCH,DIRECTORY
320 FOR J=0 TO 90:READ K$(J):NEXT
400 CLOSE 1:INPUT"NAME OF PROGRAM FILE";G$
402 PRINT:PRINT"DRIVE #: ";
404 GET C$:IF C$="" THEN 404
406 IF C$<"8" OR C$>"9" THEN 404
408 DN=VAL(C$):PRINT C$:PRINT
410 OPEN 1,DN,3,G$+",P,R"
420 GET#1,A$,B$
430 IF A$<>CHR$(1) AND A$<>"" THEN 400
440 IF A$="" THEN A$=CHR$(1):GET#1,X$
450 INPUT"LINE NUMBER RANGE  -■■■";Z$
460 L0=0:L1=0:L2=1E9
470 FOR J=1 TO LEN(Z$):Y$=MID$(Z$,J,1)
480 Y=ASC(Y$):IF Y>=48 AND Y<=57 THEN 510
490 IF Y=32 THEN 510
500 L0=J:IF Y<>45 THEN 600
510 NEXT J
520 IF L0<LEN(Z$) THEN L2=VAL(MID$(Z$,L0+1)):IF L2=0 THEN L2=1E9
530 IF L0>1 THEN L1=VAL(Z$)
540 IF L0=0 THEN L1=L2
600 P3$="[":P4$="]":PRINT:INPUT"LIST TO PRINTER  Y■■■";Z$
610 P=3:IF ASC(Z$)=89 THEN P=4:L$="@":P3$=CHR$(219):P4$=CHR$(221):GOSUB 4000
620 P1$="[":P2$="]":PRINT:INPUT"GRAPHICS OR TEXT  G■■■";Z$
630 PRINT CHR$(142);:IF ASC(Z$)<>84 THEN 640
635 PRINT CHR$(14);:M$=L$:P1$=P3$:P2$=P4$
636 PRINT:PRINT"CHANGE INTERFACE THEN PRESS ANY KEY":PRINT
637 GET XX$:IF XX$="" THEN 637
```

(continued on next page)

(continued from previous page)

```
630 PRINT"TT"
640 PRINT:INPUT"TRANSLATE CURSOR MOVES  Y███";Z$
650 IF ASC(Z$)=89 THEN T7=1
660 OPEN PN,P,F:F$=P1$:IFPN=4 THEN PRINT#P,CHR$(27);"C";CHR$(0);CHR$(10*PL);CHR$(2
7);"N";CHR$(4);
670 J=80:IF P<>3 THEN 690
680 PRINT"J":J=PEEK(213)+1
690 PRINT"JENTER DATE (DD.MM.YY)"
691 INPUT DT$
692 IF LEN(DT$)<>8 THEN 691
693 L9=J:PRINT#4,"PROGRAM   ";O$,DT$
694 PRINT"JPLEASE WAIT...."
700 REM NEW LINE
710 GOSUB 2010:Q=0:T1=1:C1=-1:GET#1,A$,B$:IF ST<>0 THEN 3000
720 IF B$="" THEN 3000
730 GET#1,A$,B$
740 L=ASC(A$+CHR$(0))+ASC(B$+CHR$(0))*256
750 IF L<L1 THEN 1080
760 IF L>L2 THEN 3000
770 F2=1:PRINT#4,M$,F$:P$=STR$(L)+" "
800 REM START TEXT HERE
810 GET#1,A$:IF A$="" THEN 710
820 T=0:A=ASC(A$):IF A=32 AND P$="," THEN 840
830 IF Q=0 OR (A AND 127)<31 OR T7=0 THEN 900
840 FOR J=0 TO 29:IF A=A(J) THEN B$=A$(J):GOTO 860
850 NEXT:GOTO 1000
860 IF B$=B1$ THEN B=B+1:GOTO 810
870 IF B>0 THEN A$=MID$(STR$(B+1),2)+P$+B$:GOTO 890
880 A$=P$+B$
890 B=0:B1$=B$:P$=",":F1=1:GOTO 1010
900 IF Q=0 THEN 905
901 IF A=92 THEN 4100
902 IF A>95 AND A<128 THEN 4100
903 IF A>160 AND A<192 THEN 4100
904 IF A>191 THEN 4200
905 A=A-128:IF A<0 OR Q<>0 THEN 1000
910 IF A=127 THEN A$="↑":GOTO 1000
920 T=1:A$=F$(A)
930 IF A=15 THEN Q=2
1000 GOSUB 2010
1010 IF A$=CHR$(34) THEN Q=1-Q
1020 REM C=-1 FOR ALPHANUMERIC
1030 C=ASC(LEFT$(A$,1)):C1=(C<48 OR C>57) AND (C<65 OR C>90)
1040 IF T<>T1 OR T=1 THEN T1=T:IF NOT C AND NOT C1 THEN P$=P$+" ":GOSUB 2500
1050 C=ASC(RIGHT$(A$,1)):C1=((C<48 OR C>57) AND (C<65 OR C>90)) OR A=37
1060 P$=P$+A$:GOSUB 2500
1070 GOTO 810
1080 REM SKIP TO NEXT LINE
1090 GET#1,A$:IF A$="" THEN 710
1100 GOTO 1090
2000 REM CLOSE OFF CURSOR EXPRESSION
2010 IF F1=0 THEN 2040
2020 IF B>0 THEN P$=P$+MID$(STR$(B+1),2):GOSUB 2500
2030 B=0:F1=0:B1$="":P$=P$+F2$:GOSUB 2500:F$=F1$
2040 RETURN
2500 IF LEN(P$)<L9 THEN 2600
2510 FOR J=L9 TO L9*.6 STEP -1
2520 IF MID$(P$,J,1)=":" THEN 2580
2530 NEXT:FOR J=L9-1 TO L9*.6-1 STEP -1
2540 P=ASC(MID$(P$,J))
2550 IF P=91 THEN 2580
2560 IF P=59 OR P=44 OR P>93 THEN J=J+1:GOTO 2580
2570 NEXT:J=L9-1
2580 PRINT#4,M$;LEFT$(P$,J-1)
2590 P$="     "+MID$(P$,J)
2600 RETURN
3000 IF LEN(P$)>0 THEN F2=1:PRINT#4,M$,P$
3010 IF F2=0 THEN PRINT"** NO LINES FOUND **"
3020 CLOSE 1:GOSUB 2000:IF PN<>4 THEN 3030
3025 PRINT#4,CHR$(12),CHR$(7),CHR$(7),CHR$(7):CLOSE 4
3030 PRINT"J████████████████████FINISHED!"
3040 END
4000 PRINT:PRINT"PRINTER DEVICE █ ";
4010 GET C$:IF C$="" THEN 4010
4020 IF C$<"4" OR C$>"5" THEN 4010
4030 PN=VAL(C$):PRINT C$
4040 PRINT:PRINT"PAPER LENGTH  1█ OR 1█? ";
4050 GET C$:IF C$="" THEN 4050
4060 IF C$<"1" OR C$>"2" THEN 4050
4070 PL=VAL(C$):PRINT"1";C$
4080 RETURN
4100 FOR J=40 TO 104:IF A=A(J) THEN B$=A$(J):GOTO 860
4110 NEXT:GOTO 930
4200 IF A>191 AND A<224 THEN A=A-96:GOTO 4100
4210 IF A>224 AND A<255 THEN A=A-64:GOTO 4100
4220 IF A=255 THEN A=126:GOTO 4100
4230 GOTO 905
```

Autodata 2

A S Burbidge,
Reigate,
Surrey.

VIC-20

AUTODATA 2 IS a short machine-code utility program designed to overcome the extremely tiresome task of converting data stored in Ram into Basic Data statements with the minimum fuss and effort.

I have found this program invaluable. It converts large lumps of memory, such as characters, into printed Data statements, leaving the user with complete control over the very good line editing facilities already available on the basic Vic-20. The program is very compact, and occupies an area of 512 bytes still leaving the user with 3K to use for storage.

In fact the program produced the Data statements in the program used to load the program itself. Using the utility, this operation took just over a minute — that would normally take over an hour, even for the fastest typist.

The program is designed for use with the unexpanded Vic-20, but can easily be relocated to any other part of Ram. It resides from 1C00 or 7168 to 1E00 or 7680, and all that needs to be changed are a few JSRs and the relocation of the interrupt vector.

Be careful when typing in the program, as it is machine code and a single mistake could crash the system. It is best to save the program before running it, or at least to check your typing thoroughly.

Once you have typed in the program and successfully run it, type Sys 7408 and Return. Nothing will appear to happen, but if you look carefully you will notice slight disturbances in your TV picture. This is because the 60Hz interrupt vector has been altered. Now all that you have to do to get your first line of data printed is to press F1.

When you hit F1, the line is printed, and you are free to edit the line as if you had just typed it, and you can hit Return to enter it as a Basic program line. Once you have done this, you can hit F1 again to get the next line printed. For example, to get data from 7680 to 8192 starting at line 100 with 14 items per line at a 10-line interval run the Autodata program and then:

```
START LINE ? 100
START ADDRESS ? 7680
END ADDRESS ? 8192
ITEMS/LINE ? 14
LINE # INTERVAL ? 10
SYS 7408, then hit F1
100 data etc.
```

You may experience inverse spaces at the ends of lines. This is due to the cursor doing weird things. ■

```
1 REM DATA FOR
2 REM AUTODATA (1C00)
3 REM SYS7408 TO
4 REM START
5 REM MC (C) A.S.
6 REM BURBIDGE 1984
100 DATA 172, 60, 3, 173, 61, 3,
 32, 145, 211, 160
110 DATA 3, 162, 68, 133, 74, 13
2, 73, 32, 215, 219
120 DATA 169, 0, 172, 67, 3, 32,
 145, 211, 160, 3
130 DATA 162, 75, 132, 74, 134,
73, 32, 215, 219, 96
140 DATA 6, 255, 0, 2, 0, 255, 0
, 255, 160, 3
150 DATA 169, 68, 32, 162, 219,
32, 12, 220, 32, 221
160 DATA 221, 162, 0, 189, 0, 1,
 240, 6, 32, 210
170 DATA 255, 232, 208, 245, 160
, 3, 169, 68, 32, 140
180 DATA 218, 160, 3, 169, 75, 3
2, 162, 219, 32, 106
190 DATA 216, 160, 3, 162, 68, 1
32, 74, 134, 73, 32
200 DATA 215, 219, 169, 68, 32,
210, 255, 169, 97, 32
210 DATA 210, 255, 96, 0, 255, 0
, 255, 0, 255, 11
220 DATA 255, 130, 255, 0, 255,
0, 255, 0, 173, 63
230 DATA 3, 172, 62, 3, 32, 145,
 211, 32, 13, 216
240 DATA 32, 221, 221, 162, 0, 1
89, 0, 1, 240, 6
250 DATA 32, 210, 255, 232, 208,
 245, 24, 173, 62, 3
260 DATA 105, 1, 141, 62, 3, 169
, 0, 109, 63, 3
270 DATA 141, 63, 3, 32, 106, 21
6, 32, 12, 220, 96
280 DATA 2, 0, 255, 0, 223, 16,
255, 0, 255, 0
290 DATA 255, 82, 174, 66, 3, 13
8, 72, 32, 48, 28
300 DATA 169, 0, 160, 0, 32, 145
, 211, 32, 12, 220
310 DATA 32, 128, 28, 104, 170,
202, 138, 72, 240, 7
320 DATA 169, 44, 32, 210, 255,
208, 239, 104, 96, 162
330 DATA 0, 189, 0, 1, 240, 8, 3
2, 210, 255, 232
340 DATA 32, 0, 28, 120, 169, 29
, 160, 0, 141, 21
350 DATA 3, 140, 20, 3, 88, 96,
165, 197, 205, 96
360 DATA 3, 208, 3, 76, 191, 234
, 141, 96, 3, 201
370 DATA 39, 208, 246, 32, 192,
28, 173, 63, 3, 205
380 DATA 65, 3, 240, 2, 16, 11,
173, 64, 3, 205
390 DATA 62, 3, 48, 3, 76, 191,
234, 169, 15, 141
400 DATA 14, 144, 169, 176, 141,
 11, 144, 76, 191, 234
410 DATA 165, 197, 205, 96, 3, 2
40, 26, 141, 96, 3
420 DATA 201, 15, 208, 255, 2, 2
55, 2, 255, 167, 144
430 DATA 0, 255, 0, 244, 128, 3,
 205, 62, 3, 48
440 DATA 8, 254, 0, 255, 0, 255,
 0, 255, 2, 251
450 DATA 0, 255, 0, 255, 0, 255,
 0, 247, 0, 255
500 FOR N=7168 TO 7680: READ A:
POKE N,A: NEXT N
```

DRIVE CUMANA

The Acorn Electron Computer

A microcomputer with a proven track record. Using BBC Basic, the Electron was developed out of the Micro that has been chosen for over 80% of schools participating in the Government's current Micros In Schools project. It connects into almost any TV set and cassette player and is supplied with a comprehensive User Guide which runs through, in a simple to follow manner, the basic principles of programming. A wide range of software is available, including games, educational packages and home accounts.

Disk interface now available for the Acorn Electron, Spectrum and Spectrum + Dragon 32/64, details from Cumana.
Electron Interface £149.95 inc. VAT.

Diskettes

Only when Cumana found a double density diskette up to their stringent quality control requirements, and at a price that made sense to the end user, were Cumana ready to put their name to it.

The result was a top quality diskette, complete with a plastic protective case for the ten-pack (they are also sold in two-packs!) together with index stickers.

Cumana diskettes are available now in both 5¼ inch and 3½ inch as single or double-sided, 40 or 80 track, from your local computer dealer. Don't trust it to chance, go for the best name on record.

Cumana BBC Microcomputer compatible disk drives:

Cumana disk drives supplied with formatting diskette, drive connecting cable and Cumana user guide. Independent power supply, mains lead and moulded plug included.

CS100 40 Track single sided (100K)	£164.95
CS200D 40 Track double sided (200K)	£169.95
CS200 80 Track single sided (200K)	£192.95
CS400 80 Track double sided (400K)	£219.95
CD200 2 x 40 Track single sided (200K)	£294.95
CD400D 2 x 40 Track double sided (400K)	£325.95
CD400/S 2 x 80 Track single sided (400K)	£359.95
CD800/S 2 x 80 Track double sided (800K)	£414.95

Cumana disk drives with independent power supply, mains lead and plug. Excluding other accessories.

CS100E 40 Track single sided (100K)	£149.95
CS200ED 40 Track double sided (200K)	£192.95
CS200E 80 Track single sided (200K)	£180.95
CS400E 80 Track double sided (400K)	£208.95

Cumana disk drives supplied with formatting diskette and Cumana user guide. Power supply taken from BBC Microcomputer, with lead supplied.

CSXL100 40 Track single sided (100K)	£115.95
CSX100 40 Track single sided (100K)	£124.95
CSX200D 40 Track double sided (200K)	£134.95
CSX200 80 Track single sided (200K)	£159.95
CSX400 80 Track Double sided (400K)	£189.95

3½ INCH DRIVES

CSX351 40 Track single sided (100K)	£139.95
CSX354 80 Track double sided (400K)	£199.95
CDX352 2 x 40 Track single sided (200K)	£256.95
CDX358 2 x 80 Double sided (800K)	£383.95

The 'X' range takes its power from your microcomputer.

CS351 40 Track single sided (100K)	£169.95
CS354 80 Track double sided (400K)	£229.95
CD352 Dual 40 Track single sided (200K)	£299.95
CD358 Dual 80 Track double sided (800K)	£429.95

The 'CS' range comes complete with power supply.

DISTRIBUTORS/RETAIL

Available from the following retail outlets: W.H.Smith, John Lewis Partnership, Greens Leisure, Laskys, Spectrum UK.

Area distributors:
Addons Ltd (Southampton) 0703 34775/6. Audio & Computer Centre Ltd (Jersey) 0534 74000. Eltec Home & Business Computers (Bradford) 0274 722512. Fairhurst Instruments (Wilmslow) 0625 525694. HCCS Associates (Gateshead) 0632 821924. Hugh-Symons Distribution Services Ltd (Bournmouth) 0202 26535. Kingdom Design (Belfast) 0232 643720. Lightning (Harrow) 01 969 5255. Microage Distribution Ltd (North London) 01 205 7688. Microworld (Edinburgh) 031 228 1111. National Micro Centre (Stockport) 061 429 8080. Northamber PLC (Chessington). Silicon Express (Leicester) 0533 374917. Thompson Cook Distribution Ltd (Birmingham) 021 328 3895. J. S. Simnett Computers (Kingston-upon-Thames) 01 541 1495.
+ National Dealer Network

CUMANA®

Cumana Limited, Pines Trading Estate, Broad Street, Guildford, Surrey. GU3 3BH
Telephone: 0483 503121

All Prices Inclusive of VAT. All 80 Track 5¼" Drives are Switchable to 40 Track Mode

All prices are subject to change without notice.

The MSX Micro that's paid its

Yamaha manufacture probably the most successful range of electronic musical instruments in the world. Their new CX5M Music Computer is no exception.

The CX5M is a fully-fledged MSX micro-computer offering the exciting advantages of its breed; an ever-growing array of standardised software, 16-colour graphics, cassette and printer interfaces, twin joystick ports and expansion slot.

But that's not all.

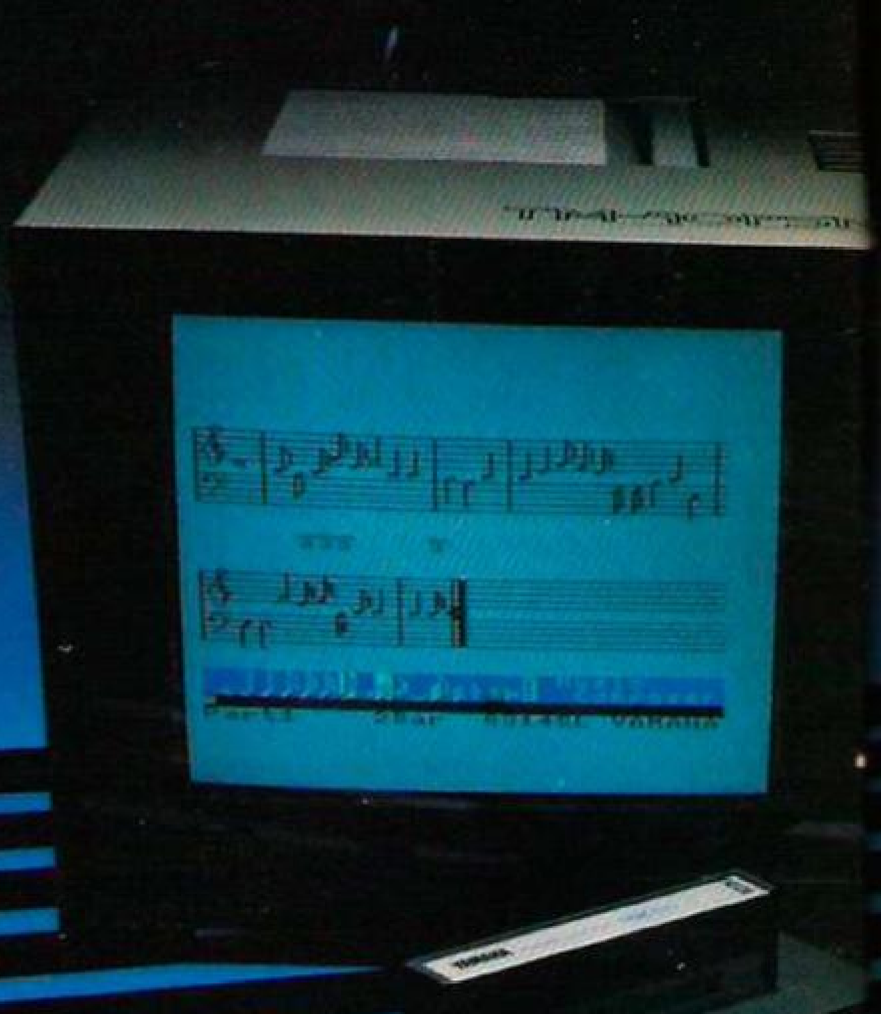
Inside the CX5 is a polyphonic, programmable FM Digital synthesiser that can be played with its own music keyboard or, via its industry-standard MIDI port, control a network of compatible musical equipment.

In 1984 Yamaha's DX Series FM Synthesizers revolutionised the voice of music synthesis with their stunning reproduction of natural and electronic sounds. Now the CX5 gives you that same musical fidelity simply by hooking-up through your TV monitor or Hi-Fi system. For the first time a computer is a real musical instrument.

Yamaha also offer a number of music-based software ROMs. Music Macro for instance is designed specifically for the computer hobbyist. It enables you to access the CX5's superb FM sounds from MSX BASIC and from this, program games and AV sequences using music and sound effects.

Or try the FM Music Composer Program which provides an on-screen musical stave for fully expressive, computer assisted composition and arrangement.

So when you've completed your modern day answer to Beethoven's piano concerto... you could always start cataloguing your record collection, work out your home accounts and discover why you have that overdraft at the bank, or even just sit back and play the latest arcade game!



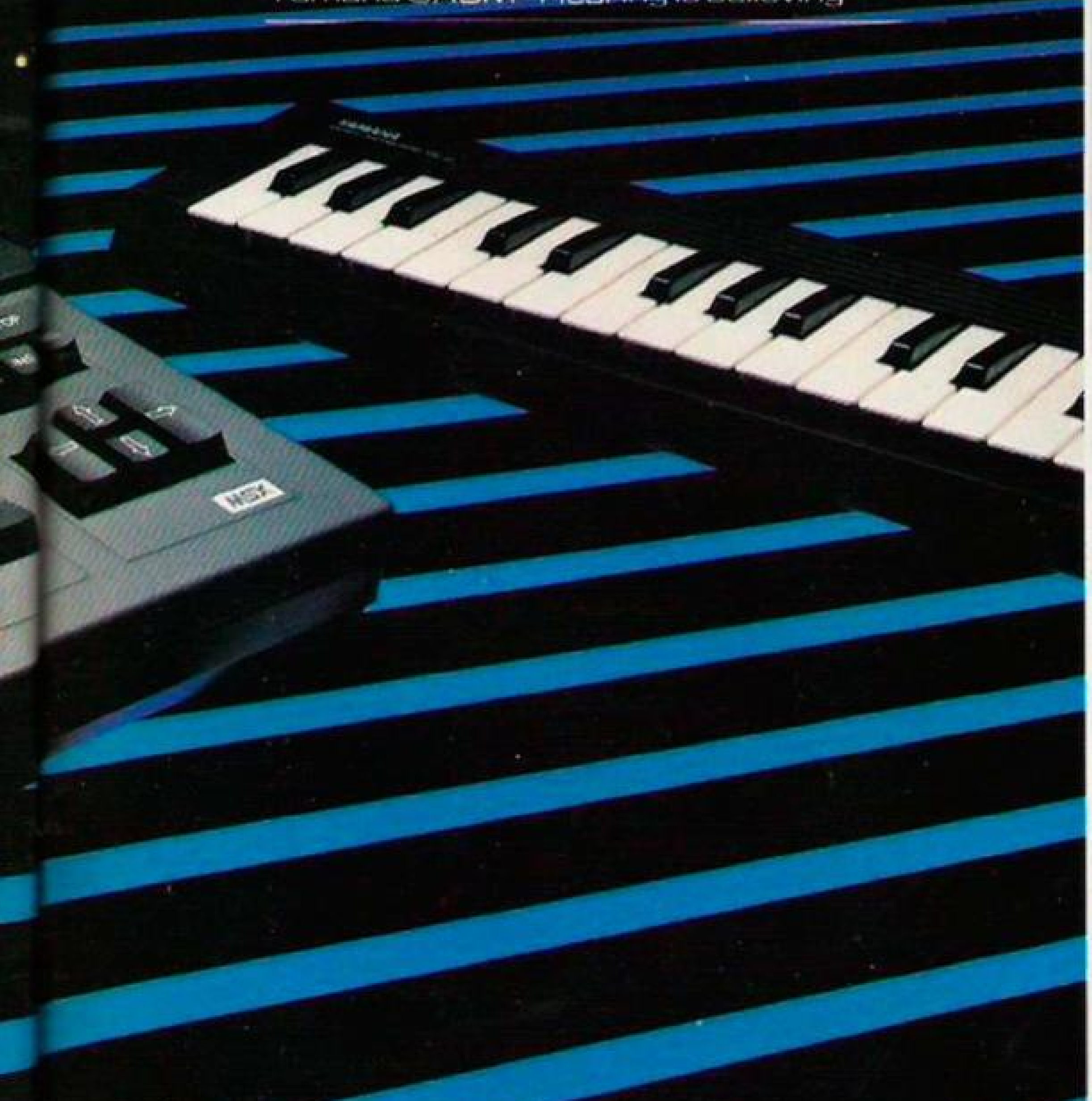


d its musical dues

Yamaha CX5M - Outline Features

- CPU - Z80A; 32K ROM; 32K RAM; 16K VRAM
- 16-colour graphics
- MIDI (Musical Instrument Digital Interface)
- Programmable FM voice generator (46-voice, 8-octave, 8-note poly)
- Music keyboard split & swap - voice and mono/poly)
- 8-voice multi-timbral
- Built-in real-time performance recorder
- Auto-accompaniment with rhythm
- Yamaha Software ROMs:
  - FM Music Macro
  - FM Music Composer
  - FM Voicing Program
  - DX7 Voicing Program (Coming Soon)
  - RX Rhythm Editor
  - 4-track Real Time Sequencer
- Price: From £534rrp (CX5M + YK01 music keyboard)
- YK10 music keyboard (full-size keys) also available

Yamaha CX5M - Hearing is believing



Yamaha CX5M - 'Hearing is Believing' FREE Demonstration tape. Fill in the coupon NOW!

See and hear the CX5M only at the following Yamaha Hi-Tech Centres.

London
Rod Argent's Keyboards, 20 Denmark St, WC2.
Chappell of Bond Street, 50 New Bond St, W1
Chromatix, Oak Rd, W5
Freedmans, 629 High Rd, E11
Future Music, 202 New Kings Rd, SW6
Gigsounds, 86-88 Mitcham Lane, SW16
Gigsounds, 22 Rushey Green, SE6
London Rock Shop, 26 Chalk Farm Rd, NW1
Rose-Morris, 11 Denmark St, WC2
Soho Soundhouse, 18a Soho Sq, W1
Syco Systems, 20 Conduit Pl, W2
Addlestone: ABC Music, 14/16 High St
Barnet: ESS, 230 High St
Belfast: Baird Sound Systems, 208 York St
Bingley: JSG Music, 104 Main St
Birmingham: Jones & Crossland, 6 Smallbrook Queensway
Musical Exchange, 89 Old Snow Hill
Blackburn: Reidy's, 9-13 Penny St
Bournemouth: Eddie Moors Music, 679 Christchurch Rd
Bristol: Bristol Guitar Workshop, 157 St Michael's Hill
London Rock Shop, 7 Union St
Cambridge: Cambridge Rock, 8 Burleigh St
Cardiff: Musicland, 148-154 North Rd
Chelmsford: Future Music, 10 Baddow Rd
Colchester: Axe Music, 96 High St
Croydon: Rockbottom, 74 London Rd
Derby: Derby Organ Centre 62 Babbington Lane
Dunfermline: Sound Control, Elgin St
Eastbourne: Peter Bonner, 12a Grove Rd
Edinburgh: James Grant, 53 Home St
Exeter: City Music, 4 Stn. Cres, Queen st
Fleet: Kingfisher Music, 20 Kings Rd
Glasgow: James Grant, 404 Byres Rd, G12
McCormacks 29-33 Bath Street G2
Guildford: Andertons, 91 Haydon Place.
Hadleigh (Essex): Honky Tonk, 300 London Rd
Harrow: City Music, 14a Broadwalk
Heald Green (Cheshire): Sounds Great, 182 Wilmslow Rd
Ipswich: Axe Music, 41-3 St Nicholas St
Leicester: Carlsbro', 22-32 Humberstone Rd
Liverpool: Frank Hessy, 62 Stanley St
Maidstone: Sharon Music, 65 High St
Mansfield: Carlsbro', 182 Chesterfield Rd Nth
Middlesbrough: Guitarzan & Bongo Bill 23/27 Middlesbrough Road
Milton Keynes: Chappell of Bond St, Silbury Boulevard.
Newcastle-upon-Tyne: Rock City, 10 Moseley St
Norwich: Carlsbro', 2 Sovereign Way, Anglia Sq.
Nottingham: Carlsbro', 11-13 Hockley
Peterborough: Stix, 603 Lincoln Rd
Plymouth: City Music, 29-31 Eastlake St
Portsmouth: Future Music, 104-106 Elm Grove, Southsea
Romford: Music Village (BAJ) 10 High Rd. Chadwell Heath
Sheffield: Carlsbro', 720 City Rd
Slough: ABC Music, 324 Farnham Rd
Southampton: Future Music, 85 St. Mary's St
Sunderland: White Sounds, 181 Hylton Rd.
Swansea: Picton Music, 9-15 Arcade
Torquay: City Music, 65 Market St
Truro: City Music, 8 Pyder St
Warrington: Dawsons Music, 65 Sankey St
Wolverhampton: One Way Music, 100 School St
Listing correct at time of going to press and subject to change

YAMAHA
MUSICAL INSTRUMENTS
MOUNT AVENUE, BLETCHLEY, MILTON KEYNES, BUCKS, MK1 1JE
(0908) 640202 24hr Ansaphone (0908) 649222

Please send me full details about the Yamaha CX5M Music Computer and your FREE demonstration tape.

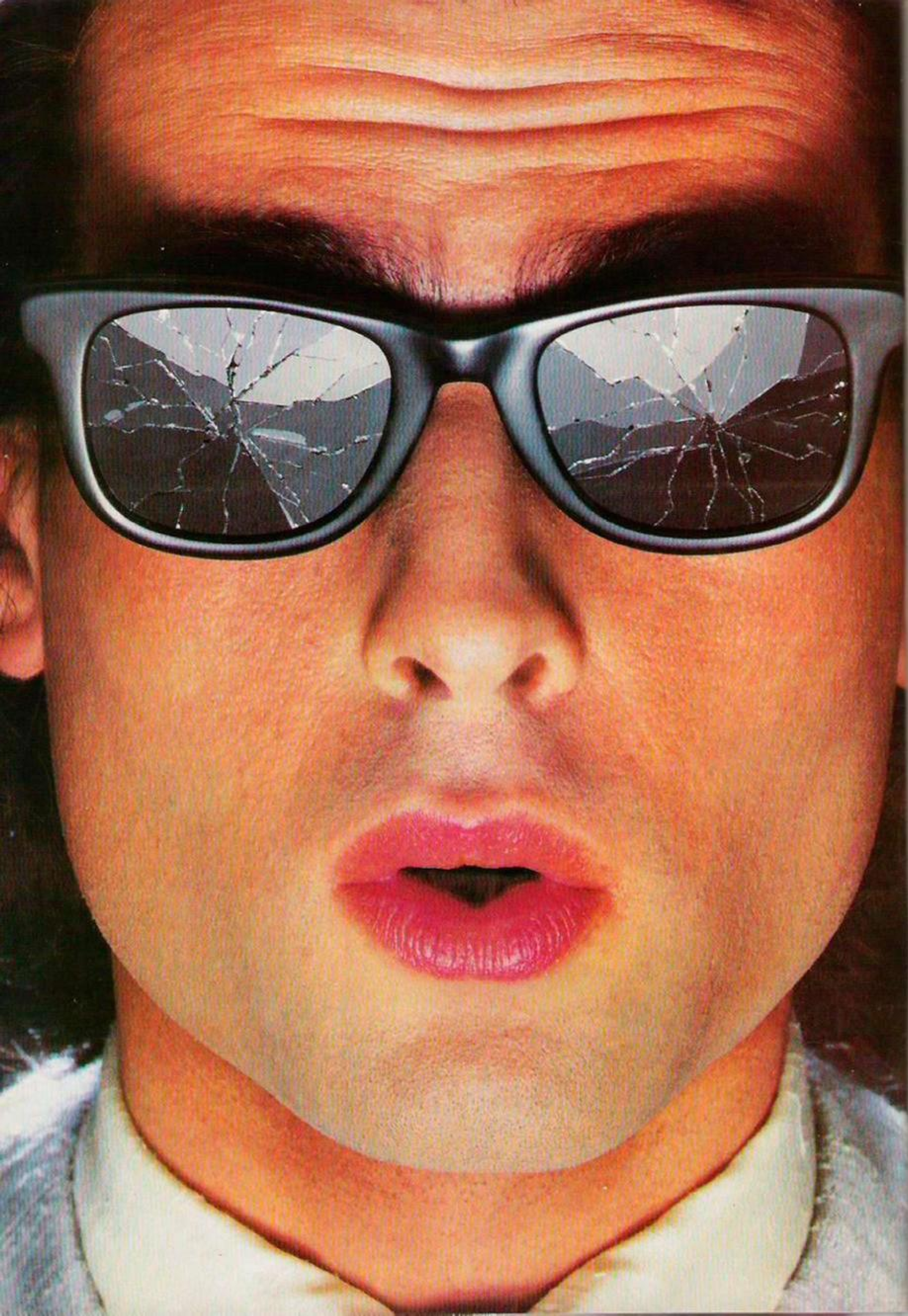
Name..............................

Address..............................

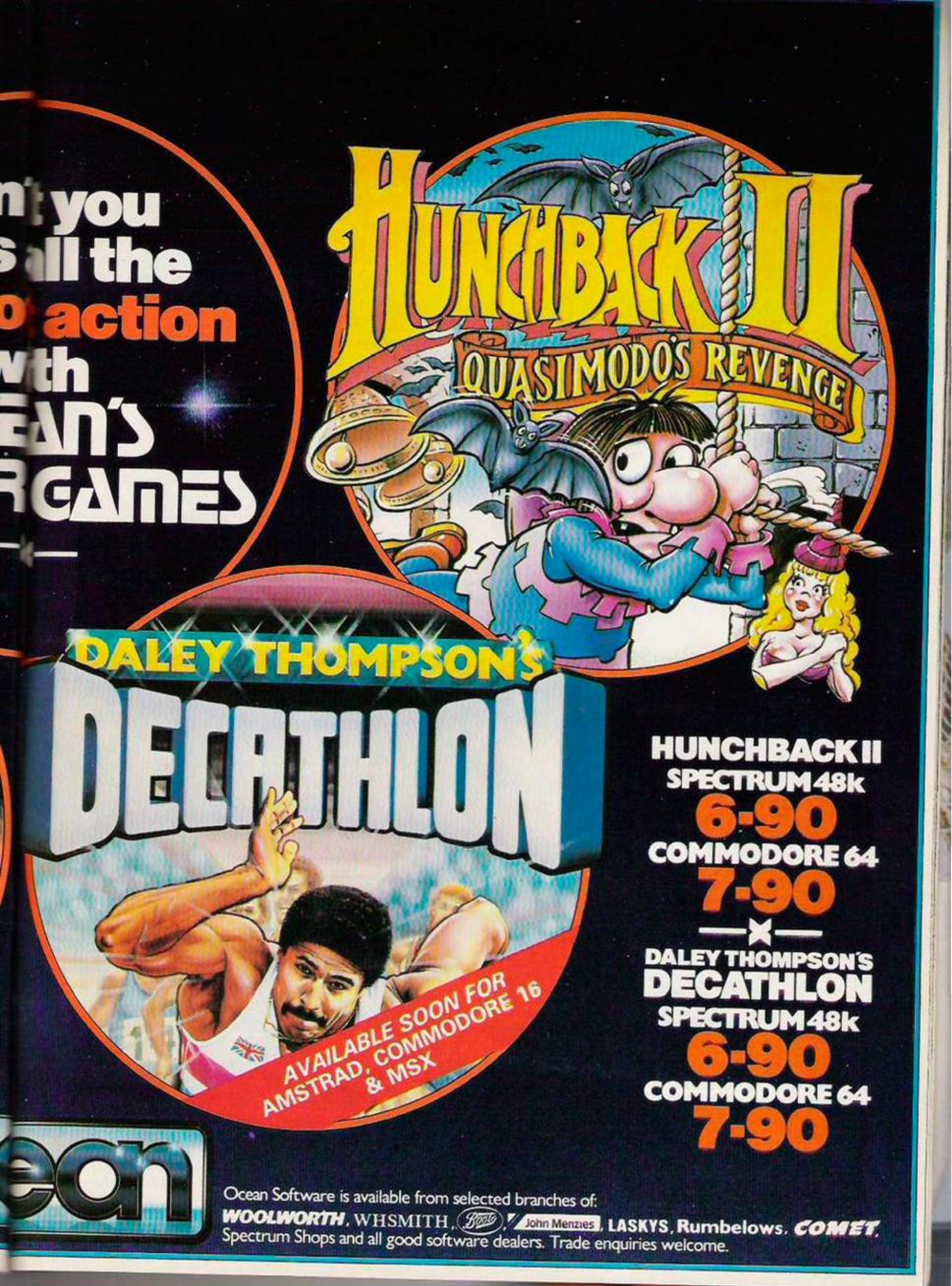
....

....

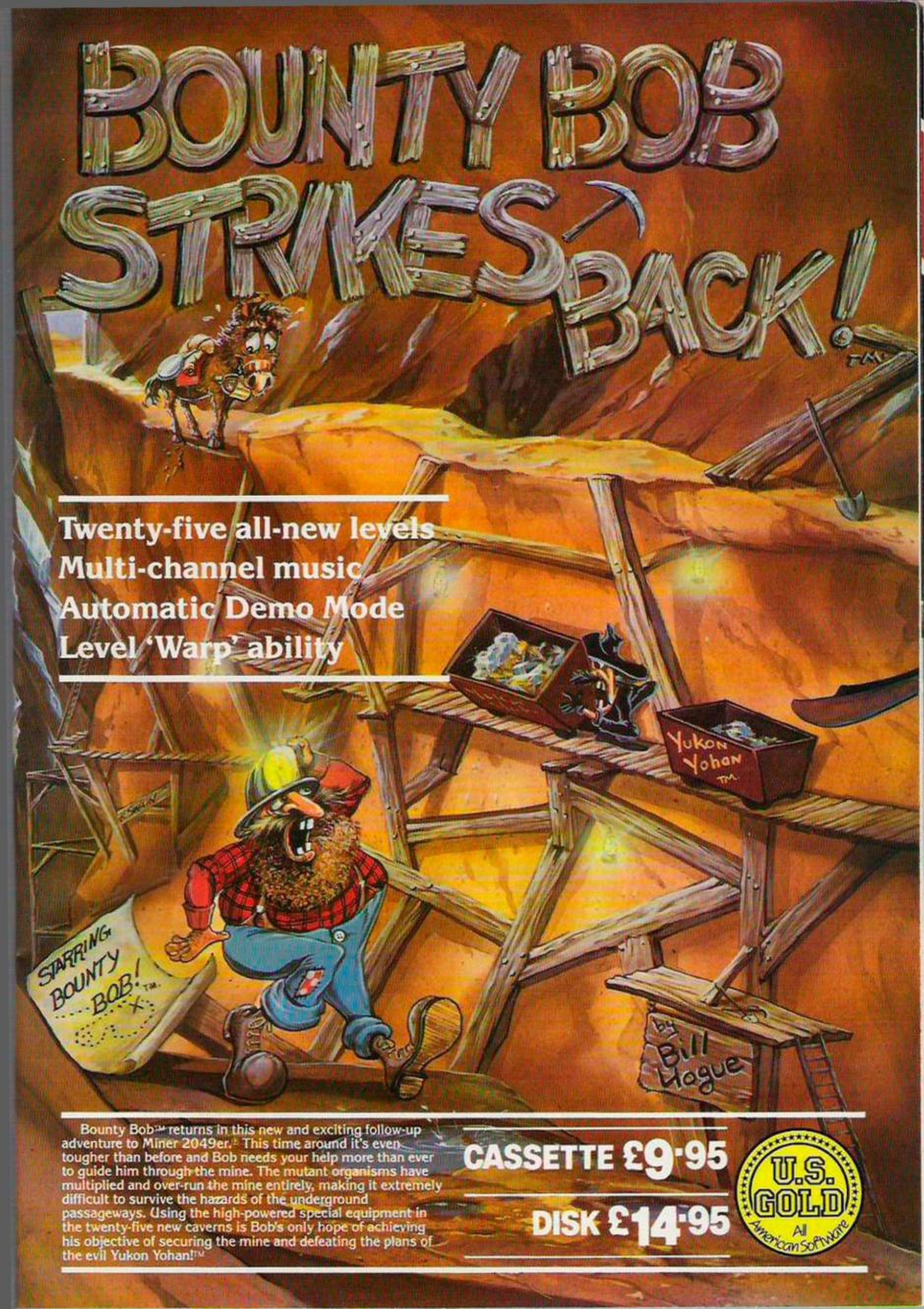
DEPT JU.1
Yamaha-Kemble Music (UK) Ltd, Mount Avenue, Bletchley, Milton Keynes, Bucks, MK1 1JE



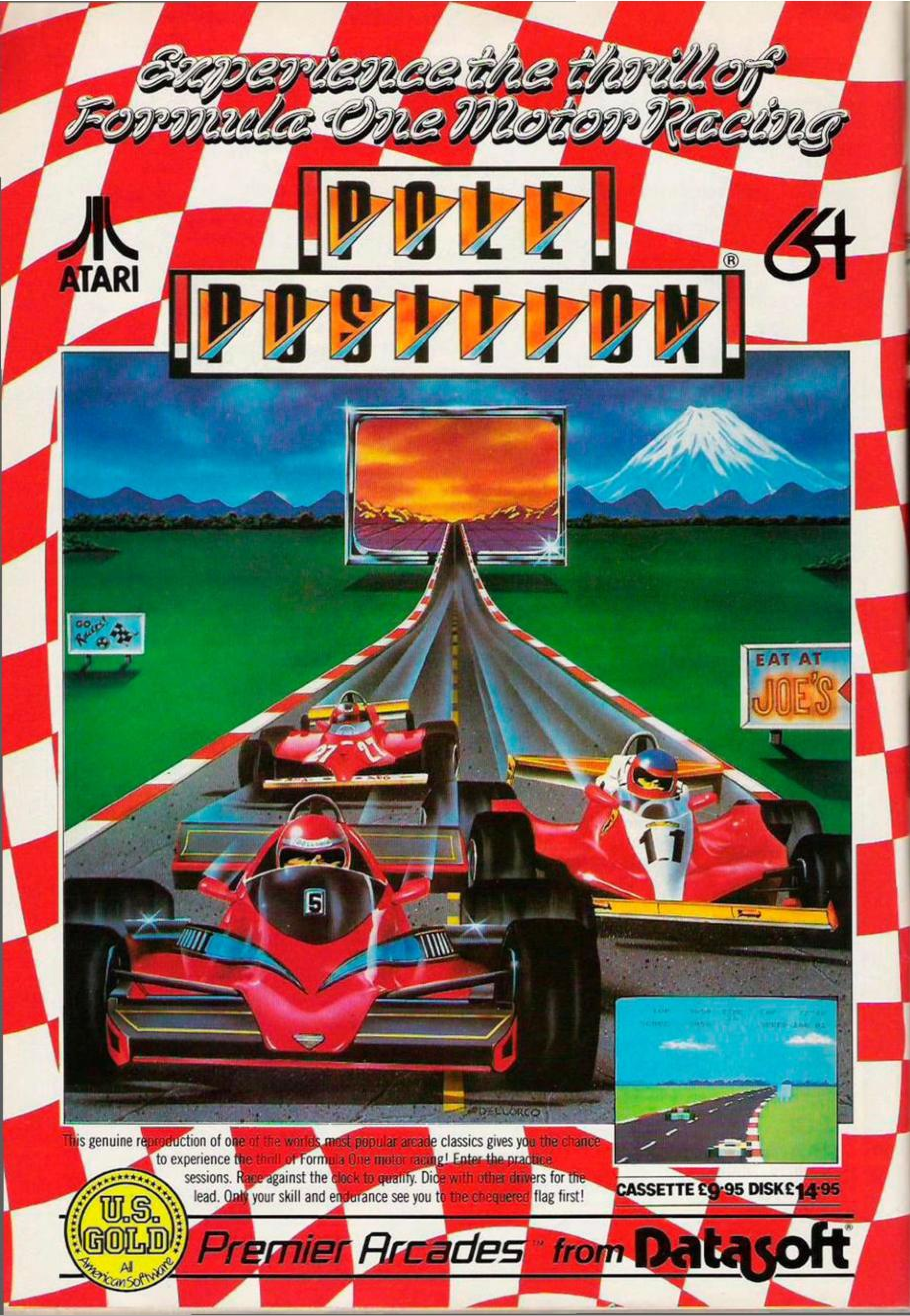

MATCH DAY
AVAILABLE SOON FOR COMMODORE 64 & BBC MODEL B
KONG STRIKES BACK!
MATCH DAY
SPECTRUM 48k
7·95
KONG
STRIKES BACK
SPECTRUM 48k
6·90
COMMODORE 64
7·90
Ocean Software
Ocean House · 6 Central Street · Manchester · M2 5NS · Telephone 061 832 6633


you
all the
action
th
GAMES
HUNCHBACK II
QUASIMODO'S REVENGE
DALEY THOMPSON'S
DECATHLON
AVAILABLE SOON FOR AMSTRAD, COMMODORE 16 & MSX
HUNCHBACK II
SPECTRUM 48k
6·90
COMMODORE 64
7·90
DALEY THOMPSON'S
DECATHLON
SPECTRUM 48k
6·90
COMMODORE 64
7·90
ean
Ocean Software is available from selected branches of:
WOOLWORTH, WHSMITH, Boots, John Menzies, LASKYS, Rumbelows, COMET,
Spectrum Shops and all good software dealers. Trade enquiries welcome.


THE ELECTRONICS MAGAZINE PROFESSIONALS CAN'T RESIST.
Electronics and Wireless World is the only electronics magazine to really tempt the professionals.
It's the only one they take the trouble to pick up for themselves.
That's because it's written for the engineer who sees electronics not only as a job, but also as a hobby.
It's written on a technical level that doesn't talk down to you.
It keeps you up to date with all the latest products and processes, applications and equipment.
And it covers every industry where electronics is involved.
Just one look at the new look magazine and you'll see why it gets professionals like you out of the office, and down to the newsagents.
ELECTRONICS & Wireless World
The leading magazine for electronics engineers.



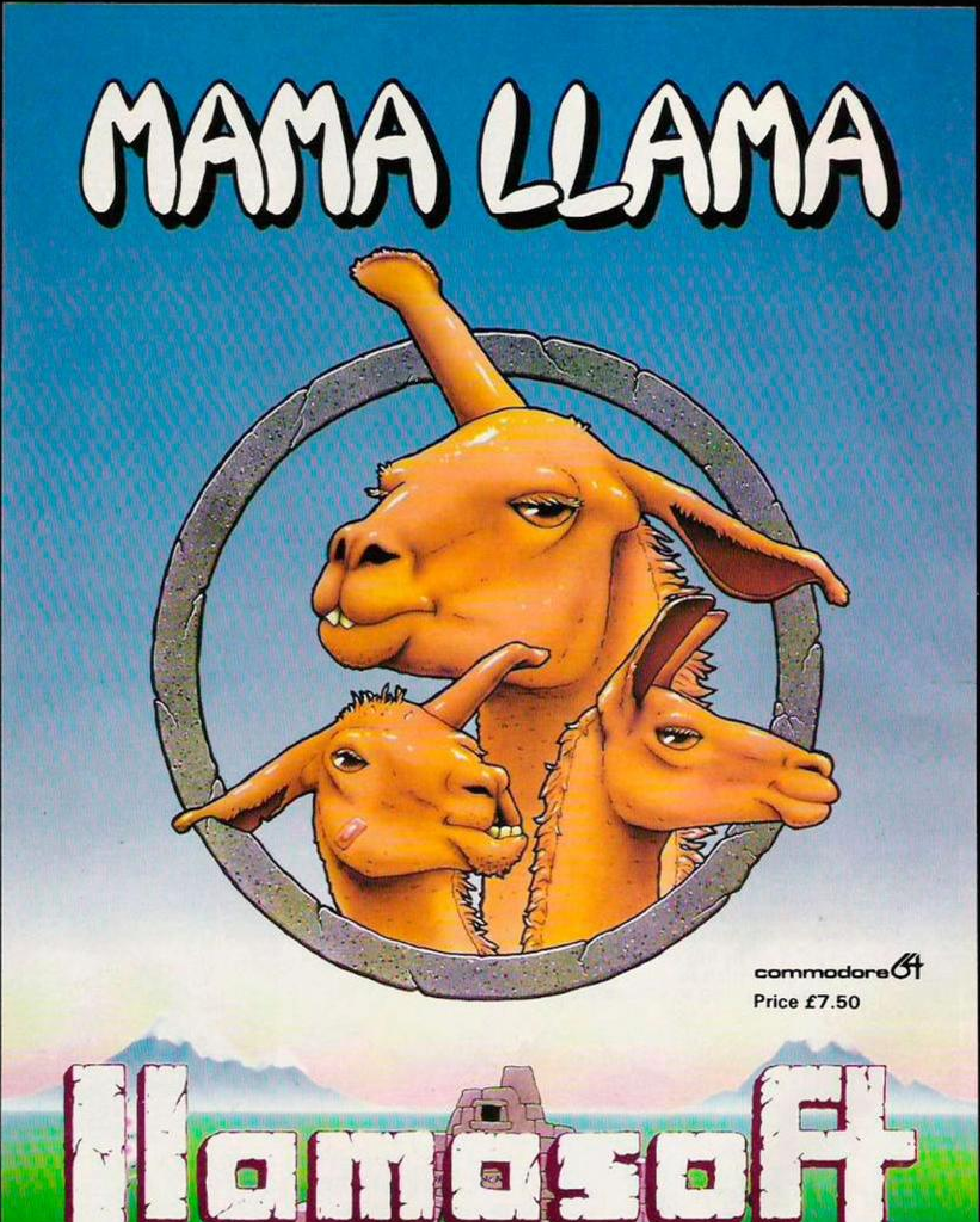
BOUNTY BOB STRIKES BACK!™
Twenty-five all-new levels
Multi-channel music
Automatic Demo Mode
Level 'Warp' ability
Yukon Yohan™
STARRING BOUNTY BOB!™
by Bill Hogue
Bounty Bob™ returns in this new and exciting follow-up adventure to Miner 2049er.® This time around it's even tougher than before and Bob needs your help more than ever to guide him through the mine. The mutant organisms have multiplied and over-run the mine entirely, making it extremely difficult to survive the hazards of the underground passageways. Using the high-powered special equipment in the twenty-five new caverns is Bob's only hope of achieving his objective of securing the mine and defeating the plans of the evil Yukon Yohan!™
CASSETTE £9·95
DISK £14·95
U.S. GOLD
All American Software



Experience the thrill of Formula One Motor Racing
ATARI
POLE POSITION®
64
GO Racers!
EAT AT JOE'S
This genuine reproduction of one of the worlds most popular arcade classics gives you the chance to experience the thrill of Formula One motor racing! Enter the practice sessions. Race against the clock to qualify. Dice with other drivers for the lead. Only your skill and endurance see you to the chequered flag first!
CASSETTE £9·95 DISK £14·95
U.S. GOLD All American Software
Premier Arcades™ from Datasoft®



'Engines on...check...
flaps to 10 degrees...
check...increase
throttle, release
brakes. Increase
throttle to full power
...watch air speed
indicator. Aircraft
taxiing, increasing
speed...170 knots...
rotate, observe
vertical speed
indicator...
increase elevator
angle.
Take off achieved!'
The excitement of a DACC 747 take off fires the adrenalin like no other computer simulation. The 3D pilot's view and full array of instrumentation challenges you to take the role of piloting a Jumbo Jet...and once in control and taxiing down the runway, there's no turning back.
Take off achieved. Full flying control at your fingertips and seven British Airports to fly to...the drama of landing is waiting to unfold as you sit in the hot seat of a 747 flight deck.
visible in true perspective on take off and landing. Optional use of joysticks.
100% Machine Code on Commodore 64k and Spectrum 48k versions.
Also available for Dragon 32/64, Tandy C/C (32k), BBC Model B, Acorn Electron, Atari 400/600/800 (48k).
Cassette Price £9.95 including VAT and postage. Spectrum Price £7.95. Commodore on Disk Price £11.95.
747 FLIGHT SIMULATOR FROM DACC
The leaders in Flight Simulation.
dacc
LIMITED
To: DACC Ltd. (Dept. YC4),
23 Waverley Road, Hindley,
Wigan, Lancashire WN2 3BN.
Name (BLOCK LETTERS)
Address
County
Postcode
Please send me
copies of the DACC 747 Flight Simulator
at
I enclose cheque/Postal Order for
the total of
My computer is a
You may order by Access, either by post or 24hr. Telephone (0942) 522002
Despatch within 48hrs.
Please debit my Access account
Access
All dealer enquiries to sole distributor: MICRO AIDS, Beech House, Hob Hey Lane, Culcheth, Warrington.
Tel: 092 576 2613/5930.


MAMA LLAMA
commodore 64
Price £7.50
Llamasoft
ORIGINAL SOFTWARE DESIGN


Cute.
Clever.
Mischievous.
Intelligent.
Dangerous.
GREMLINS

YOUR COMPUTER CLASSIFIED

01-661 3036

Contact: Susan Platts

ADVERTISEMENT RATES — Copy Date April 29th

Lineage: 50p per word (Min 20 words)
Box No: £4.00 extra
Linage advertisements are prepayable and the order form published in this section should be completed and returned with remittance. Credit Card facilities are available.

Display — rates per scc (Min. 2scc)

One insertion	:	£9.00
Three insertions	:	£8.70
Six insertions	:	£8.50
Twelve insertions	:	£8.00

Display advertisers should preferably reserve space by phone.

Closing date for Classified advertisements is the 29th April for the June edition

Post to
Your Computer, Classified Department, Room H211, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS.

Britain's Biggest Selling Home Computer Magazine

WDSoftware
Utilities for the QL (£10 on Microdrive)
This package offers single-key loading, running or deletion of files, multiple formatting of cartridges, and auto backup of files from one or more cartridges to another with optional over-writing of namesakes. In addition, a directory of 60 + files can be viewed without screen overflow. NB a VERIFY program is unnecessary — QDOS does this automatically. Now includes directory print-out and 'qprint' to print any or all files on a cartridge with optional dated titles. Owners of the earlier package can update by sending OUR cartridge and 25p.
Morse Tutor for QL £8.50
As for 48K Spectrum without speech
Current Spectrum Software
Tradewind (£5 Cassette)
Buy a ship, provisions and cargoes to trade between the harbours of the Coral Islands. Make a fortune if the weather and pirates let you live long enough to master seamanship and economics. A strategy/-adventure game with graphics for 48K.
Jersey Quest (£5 Cassette)
Following an accident, you are stranded in the labyrinth of time and must find your own way back to the present with the aid of persons and objects from Jersey Folklore. Meet the Black Horse of St Ouen, the Dragon and Knight of La Hougue Bie, the Black Cat of Carrefour a Cendre, the Witches of Rocqueburg and many others. A text-only adventure for 48K in machine code (Quill).
Morse Tutor (£4 Cassette)
The one has it all! 4-19 words/minute, variable spacing, variable number of groups of random letters numbers or mixed, random sentences, own message, single characters (including punctuation), variable pitch, FEEDBACK via screen, printer or SPEECH (phonetic alphabet via Currah u/Speech on 48K version only), and repeat facility. Takes you to and beyond RYA and Amateur Radio receiving standards. 16K and 48K versions on one cassette. 48K only on Microdrive cartridge.
Spectrum Programs on Microdrive
All the above are available on Microdrive cartridges at a price of £4.50 for one cartridge + the price of each program as shown above. E.G., a cartridge with Tradewind and Morse Tutor would be £13.50. As a bonus, Gavin Monk's Indexed Loader is included free with each cartridge.
available from
WDSoftware,
Hilltop, St. Mary, Jersey, C.I.
Tel (0534) 81392
558

AMSTRAD
AMSTRAD OWNERS
Join the original and best user club for the CPC-464.
'AMSCLUB': Look what we have to offer: Monthly tape magazine, news, reviews, programs, software/Hardware discounts, exchange club and much more. We already have members from all over Europe (France, Belgium, Spain).
JOIN THEM, JOIN US.
1 Year's membership £3 (overseas £5). Send cheques/PO to:
AMSCLUB, 42 MOUNT STEWART STREET, CARLUKE, LANARKSHIRE PL93
618

REFORM SOFTWARE
announce
PAGEWORD and PAGECODE
DIRECT REPLACEMENT modules for the Tasword Two BASIC program. No changes to the machine code are required.
PAGE/UNPAGE text; page NUMBER with TOTAL option; multiline independent HEADERS and FOOTERS; OPEN out/CLOSE up lines; SAVE/INSERT/DELETE a block; LOAD/SAVE to/from microdrive or cassette NORMAL/ALTERNATE printer control code multiple COPIES; RECONFIGURABLE to suit most user requirements.
Supplied on cassette with instructions (or send cartridge for return) £9.99
REFORM SOFTWARE
4 Evelegh Road
Farlington Hants
PO6 1DL
630

STOP—REVEAL
A unique cassette for the 48K Spectrum. Genuinely lets you:
1. STOP virtually all programs, BASIC, MACHINE-CODE and even HEADERLESS FILES, protected or not.
2. REVEALs the hidden Basic in Machine-Code programs. (Could be clues in adventure games).
3. LISTS actual Machine-Code on your video.
Now you can STOP, LIST, MODIFY, STUDY most Spectrum programs at your leisure. (And cassette is Microdrive compatible).
Includes a DISSASSEMBLER so can even study Sinclair ROM, etc, plus the most Advanced HEADER READER ever written, plus pages of SECRETS OF THE TRADE explaining professional methods used to hide lines/programs, etc.
YOU CANNOT GET A MORE POWERFUL CASSETTE
(Even STOPS/REVEALS programs which are a genuine 48K long).
POST £5.95 FREE
(Europe add £1. Other areas add £2)
KERR, 1-2 HIGH STREET, KING'S LYNN, PE20 1BX. 332

MEMOTECH
BRUNWORD £19.50 inc. Simply the best! User club — "I was amazed how much power the programmer has managed to cram into the tape package . . .". Loads in 31 sec. Files in memory, cut & paste, search, block delete etc.
DATAFILE £13.50 inc. New release. Powerful card filing system. Very fast alphabetical, numerical or date sort. User headings & print formats.
NEWORG £5.00 inc. New release. Relocate your assembly programmes.
(Overseas add £2.00 per order)
SAE for details or cheque to:
BRUNNING SOFTWARE
34 Helston Road, Chelmsford, Essex CM1 5JF. 0245 252854 607

SPECIAL AGENT
DE-LUXE SPECTRUM COPIER

A multi-purpose yet extremely easy to use tape copier. Ideal for the experienced or absolute beginner.

****** 5 STAR RATINGS ******

1. Copies multipart programs all in one loading.
2. Copies Basic, M/Code and even the Headerless programs upto 51K.
3. Cannot be fooled by weird names or false Headers. (Other copiers would crash).
4. Can remove Autostart from Basic.
5. You can determine Pauses between sections if desired.

Once you've experienced this super DE-LUXE silk smooth copier, you'll never want to use any other.

£5.95

O/seas, Europe +50p
Others +£1.50
(Order as SPECIAL AGENT)

KERR, 24/25 High Street
King's Lynn, Norfolk, PE30 1BP
511

THE ULTIMATE Spectrum Tape Back-up. You've seen the rest now buy the best. Menu driven with on screen header information. In decimal and hex only £6.95 inc p&p. ASK Software, 25 Ffordd Ffynnon, Hawarden, Clwyd CH5 3JU. 629

AMSTRAD
Super deluxe tape copier. Copies all your Amstrad tapes at your selected speed. Removes protection.
£5.95
Please order as AM1
O/seas Europe +60p
elsewhere +£1.60
ZX Guaranteed
29 Chadderton Drive, Unsworth, Bury, Lancs. BL9 8NL. 627

*** BACK-UP SYSTEMS FOR ***
*** CBM-64 & VIC-20 ***

NEW! BACKUPBOARD Mk II NEW!
This new version of our famous Back up Board now works two ways. Switched to mode 1 it operates as before, i.e. you can connect a second CBM cassette deck to your VIC or 64 and make a copy of a program as it loads. It will copy m/c code, fast loaders, data files etc. By switching the board to mode 2, '64' owners can use our special transfer software to regenerate a copy of a program on the second deck as it loads from the first. This method can produce a better quality copy than the original.
Conversion service. Return your original Back up Board to be modified to Mk II for £4.50.
Back up Board Mk II with software on cassette. £14.50 inc p&p
Back up Board Mk.I (Operates in Mode 1). Still only £10.00 inc p&p
Send your cheque/PO to
TURBOTRONIC
46 Ripon Street, Parkinson Lane, Halifax, West Yorkshire HX1 3UG.

MULTIPLY your income/profits NOW with the breakthrough ENTREPRENEUR ASSOCIATION. Discover new, lasting success. Act today! and send for full, free details: ENTREPRENEUR ASSOCIATION, Dept A4, 9 Cork Street, Mayfair, London W1. 473

DRAGON Mediacalc Spread Sheet 756 cells +-/*%»». Average total R&Col's. All functions supported with menu and instructions. Only £9.99. Media Magnetics, Freepost, West Bromwich B70 6BR. 596

ASTROLOGY
for beginners
only £11.50
A Starter Pack comprising a simple program to calculate a horoscope, an introductory booklet & 2 self-teaching programs (how to interpret the horoscope).
No previous knowledge required
48K Spectrum, BBC/Electron, 48K Oric, Dragon, Tandy Color, Tandy TRS80, Commodore 64/PET, Atari, Amstrad, Sharp MX80A/K/700, Colour Genie, Video Genie/System 80/PMC 80
many more programs for experienced astrologers
also other micros
Dez programmas zijn thans in Nederlandstalige versies verkrijgbaar voor Spectrum. Vraag Documentatie! Cash with order (add 50p outside UK) or large sae for the free catalogue to:
ASTROCALC (Dept YC)
67 Peascroft Road
Hemel Hempstead,
Herts HP3 8ER
Tel: 0442 51809
537

! 1985 VERSION!

Would you like to transfer your Spectrum programs from

① MICRODRIVE TO MICRODRIVE ③ TAPE TO TAPE
② TAPE TO MICRODRIVE ④ MICRODRIVE TO TAPE

Then do it with

TRANS-EXPRESS

TAPE TO MICRODRIVE — MICRODRIVE TO TAPE

TRANS-EXPRESS is the most comprehensive package of four m/c utilities for transferring Spectrum programs. They are user-friendly, simple to use, reliable & very efficient.

They will enable you to transfer any kind of programs up to the full 48.0K length. TAPE TO M/DRIVE will also automatically, if necessary, modify, restructure and relocate both BASIC and M/C to run from m'drives.

You can buy any of the 4 programs for £5.50 or TAPE TO M/DRIVE AND M/DRIVE TO M/DRIVE together for £7.50, or all the 4 programs on a tape for £9.95 or on a m'drive cartridge for £14.95.

MICRODRIVE TO MICRODRIVE

NEW! . . . AMSTRAD version of **TAPE TO TAPE** at £5.95 only. **NEW! . . .** Transfers programs of any length, any speed, headerless, with unorthodox loaders, etc. Continuous LOAD/SAVE, repeated SAVE, full abort., full header, reader, unprotect, etc.

Please send me a copy of **TRANS-EXPRESS** ①②③④ Please tick where applicable I enclose cheque/PO for .... YC
(Please add £1.00 for orders outside UK)
Name ....
Address ....

ROMANTIC ROBOT 77 Dyne Rd. London NW6 7DS 108/108

BLANK CASSETTES

Rely on the professionals for guaranteed top quality computer/audio cassettes at great value budget prices. Packed in boxes of 10, complete with labels, inlay cards and library case.

Prices include VAT, post & packing

Length	Box Price (10)	Qty.	Value
5 mins (c.5)	£4.35		
10 mins (c.10)	£4.40		
12 mins (c.12)	£4.45		
15 mins (c.15)	£4.50		
30 mins (c.30)	£4.70		
60 mins (c.60)	£5.30		
90 mins (c.90)	£7.00		
			£

Cheque Postal Order enclosed for

TRADE ENQUIRIES WELCOME. GET BEST TERMS, SERVICE & PRODUCT, BUY DIRECT FROM MANUFACTURERS OF **PROFESSIONAL** BRAND CASSETTES.

NAME ....

ADDRESS ....

....

PROFESSIONAL MAGNETICS LTD
(Dept. CCO), Cassette House, 329 Hunslet Road, Leeds LS10 3YY
FREEPOST Tel: (0532) 706066

400

COMPUTER REPAIR CENTRE
Have you got problems with your computer?
THEN CALL US
BBC, Spectrum, CBM, Dragon
Mail order or call in
ENFIELD COMMUNICATIONS LTD
135 High St, Ponders End, Enfield, Middx.
01-805 7772

624

C.64 SWAP
(A division of Cynthatronics International)
Tape exchange club for Commodore 64
* Free Membership * Free Exchange
* Free Postage * Latest Games
* 2 (or more) for 1 swap (limited period)

11 Cynthia Road, Poole, Dorset.
(0202) 731594

626

MICRODRIVE BACK-UP
"THE PRINCE"

Transfers all ultimate games including "Underworlde" and "Knight Lore". Transfers "Decathlon", "Cyclone", "Ghostbusters", "Lords of Midnight", "Monty Mole", "Codename Mat", "Trashman", "Pygamarama", "Scrabble", "Chequered Flag", most imagine games and many more. Absolutely guaranteed to work! Incredible value at £6.95

Cheques to: **M. Alexander, Alexander Computing Systems, 67 Lansdowns Lane, Charlton, London SE7.**

Alexander Computing Systems 1985

593

Spectrum Repairs

£16.95 inclusive of labour, parts, p&p etc. ZX81 for £11. Fast reliable service. (Av. repair 24 hrs) by qualified engineers. Send computer only to:
HS COMPUTER SERVICES
2 The Orchard, Warton, Preston, Lancs. PR4 1BE. Tel. 0772 632686 or 634897

327

MICRODRIVE-WAFADRIVE TRANSFER TAPES

Transfer majority of your games to WAFADRIVE or MICRODRIVE using the 007WD or 007MD tapes. These use the Semi-Automatic method proved to be the MOST successful available. Includes transferring of Headerless programs & programs in map area.

The Wafadrive 007WD includes system to greatly reduce Loading time for multipart progress.

Longer programs/Hyper Loads can be transferred to WAFA/CARTRIDGE using techniques printed in Newsletter of MICRODRIVE & WAFADRIVE EXCHANGE. Scrabble, Full throttle, etc, etc.

007WD or 007M on cassette £5.95
007WD on WAFA £8.95
007MD on CARTRIDGE £7.95

O/seas. Europe +55p Others +£1.55

MICRODRIVE/WAFADRIVE EXCHANGE CLUB
Membership is just £6 for 12 issues (£9 Europe, £12 elsewhere)

ZX-GUARANTEED (SPU)
29 Chadderton Drive, Unsworth, Bury, Lancs, BL9 8NL.
Tel: 061-766 5712

The Publishers take all reasonable care to ensure that classified advertisements are genuine, but readers must satisfy themselves that they will be obtaining what they require before entering into transactions, particularly if they involve large sums of money.

128/128

☆ UNLOCK YOUR AMSTRAD ☆
with AMSKEY

Easy to use, utility program removes protection and allows listing, studying and copying of your precious software. You choose, loading speed and protection of your back-up copies. Fully automatic, header reader with on-screen instructions and prompts.

FREE PEAK AT PROGRAMME. Allows you to find the messages and words in your machine code games (very useful for adventures).

Only £6.99 including p&p

Overseas orders please add £1 postage

Interlock Services Ltd.
Dept YC
37B New Cavendish St.
London W1M 8JR.
Tel: 01-609 8301

614

NEW — Commodore 64 —

TURBO-BREAKER

At last, Transfer Turbo games to Disk. Blank screen, Flashing and picture screen turbos by most software houses. Easy to use...............£10.95

•••••••• TURBO-SAVER ••••••••
Make easy Turbo back up copies of your slow load games, no user knowledge needed, easy to use...£8.00

•••••••• TRANSFER — MK2 ••••••••
Transfer most slow load games to Disk. Simple to use, and needs no user knowledge, or skill...............£8.00

•••••••• DISK-TURBO ••••••••
Make your disk games load faster, 45secs instead of 2mins, does no harm to your drive...............£8.00

•••••••• DISKOPY ••••••••
A Selective disk copier, fast and very easy to use...............£7.00

•••••••• DISK-DOCTOR ••••••••
A Disk editor, alter any part of the disk, recover a scratched file...............£8.00

ARROW MICRO SERVICES
20 Portmeadow Walk
London SE2

590

WORLD Series Cricket. Captain England against West Indies or World XI. Includes speech. 48K Spectrum. £4.95. Bosoft, Lynbrook, Borth, Dyfed. 604

SOFTWARE exchange. Swap your used programs. FREE membership. Spectrum/Dragon. Huge program base. SAE for details: UKSEC(YC), 15 Tunwell Greave, Sheffield, S5 9GB. 602

"ACCOUNTS", 48K Spectrum, maintains cash and bank accounts, analysing 1,000 entries under 26 headings in full detail, plus trial balance. £3.75. Also "Address" holds 200 addresses £2.50. Cheques/PO to Lunn (Accounting Services) Box 621. 621

ZX81 Learner Labs. £10.00. ZX81 64K RAM packs £25.00, ZX81 32K RAM packs £15.00, ZX81 16K RAM pack £10.00. Lenson 061 442 6041. 619

MAGIC-MESSAGE

Put YOUR name in Machine-Code game or simply alter existing messages. Could also reveal hidden words in Adventures. All this is possible with MAGIC MESSAGE which lets you look at and change messages in a Bytes or "disguised" Basic program. Even Commercial and the Headerless programs can be changed.

Load MAGIC MESSAGE, play in program & like Magic messages will appear on your video.

Post £5.95 Free
O/seas, Europe +55p
Others +£1.55
(Please order as Magic M)

ZX-GUARANTEED (SPU)
29 Chadderton Drive, Unsworth, Bury, Lancs, BL9 8NL.



STEVE JACKSON'S
SWORDMASTER
SERIES

The definitive role playing adventure from the pen of the world's leading author and the keys of the world's leading software house.

In your High Street shortly.

Adventure INTERNATIONAL ™

85 New Summer Street, Birmingham B19 3TE Tel: 021-359 0801

585

SINCLAIR Service Centre

We have the spares for all repairs to: Spectrum, ZX81, Interface 1, Microdrive, Ram Pack.
You pay £10.00 and parts and return postage. All units repaired to the original standard using genuine Sinclair spares by our HNC qualified staff.
Also available Spectrum memory update 16K-48K fitted by us £28.50.

Quantum, 33 City Arcade, Coventry (0203) 24632

565

ORIC

ADVANCED USER GUIDE

For the Oric-1/Atmos, this book gives a complete description of the operation of these computers, plus a complete ROM disassembly and circuit diagram.

287 PP. ISBN 0 947929 01 0

Available from bookshops or by mail at £10.45 (inc. p&p) from:

ADDER PUBLISHING LTD., PO BOX 148, CAMBRIDGE CB1 2EQ.

578

Remploy Metal Furniture — can you keep your computer on anything less?

For further information on our computer work station and display/storage module, please write to: Cass Richards, Remploy Metal Furniture, Neath Abbey Road, Neath, West Glamorgan, SA10 7BL. Telephone 0639 3812, Telex 48229.



Remploy

AMSTRAD SOFTWARE in ROM!

MAXAM

Meet MAXAM — a new full-feature no-compromise Assembler/Disassembler/Editor — with a difference. It's in a very full 16K EPROM which plugs directly into the AMSTRAD. No waiting while it loads — it's always there. You can still use the Disc unit. You also get, as a bonus, a new expansion socket for Arnor's range of Sideways ROM cartridges (containing, for example, our forthcoming Word Processor).
MAXAM uses no BASIC RAM space. It lets you mix BASIC and Machine Code — just like the BEEB. Or, you can assemble direct from the Editor, and you can even use the Editor to edit BASIC programs.
MAXAM is ESSENTIAL software for the serious AMSTRAD user.



MAXAM, including Adaptor: £59.90. On Cassette (simplified): £13.50. On Disc (full-feature): £26.90. Prices include post/packing. Full handbook supplied.
Order Hotline 01-653 1483 (2pm-6pm).

Technical Data

*Super-fast 3000lines/min assembly. *Conditional Assembly. *Plain English error messages. *Full Expression evaluation. *Unrestricted lable names. *Directives include: ORG, BYTR, WORD, TEXT, RMEM, LET, IF, GET, PUT, LIMIT, CODE, NOCODE, READ. Commands include: LIST, NOLIST, LISTP, TITLE, PAGE, PLEN, WIDTH, DUMP.

*Menu-driven Screen Editor includes move copy and delete block, tabs, search and replace, print all/part of text, Load/Save all/part of text. Disc/ROM version only: Register display, Memory Edit commands, breakpoint, string search in RAM. Link to AMSDOS.

Cheques/Postal Orders to:
Arnor Limited, PO Box 619, London SE25 6JL.
Technical Enquiries: 01-852 2174

Arnor LTD

592

POWERFUL UTILITIES

AMSTRAD SPEEDMASTER FAST LOAD UTILITY

SPEEDMASTER is a tape copier with a **difference.** It allows you to save the copy at **a choice of three speeds**, normal, fast or very fast. Also incorporates a **header reader** and **deprotect basic** option. Continuous load and save. Tests have proved **100% successful**. An **invaluable** piece of software.

ONLY £7.95

Tape to disc utility due shortly

TAPE COPIERS

Unique machine code programs to allow security back-up copies of the majority of protected software.

Available for:

Commodore 64	£5.95
VIC 20 (All memory sizes)	£5.95
Spectrum	£5.95
Electron	£5.95
Atari (All models)	£5.95
BBC (handles locked files)	£5.95

All with full instructions

TAPE TO DISC UTILITIES

BBC
A **very useful** utility to help shift those stubborn tape programs onto **your own discs** **ONLY £9.95**

ATARI
A dual purpose program that allows single stage boot programs to be shifted onto disc. Includes output to tape. Works on **all models**

ONLY £9.95

COMMODORE 64

THE BEST RANGE OF UTILITIES FOR THE 64, WRITTEN BY PROFESSIONALS – SOLD BY PROFESSIONALS

DISC DISECTOR V2.0

At last a **really powerful** disc copier and utility program. Incorporates a range of powerful programs, includes **"Fastcopy"**. Takes just **4 minutes** to read and write a **whole disc**. **"Error Copy"** automatically reads and writes **all** error types, now you can make a security back up of those expensive business programs. **"Quickdisc"** more than **halves** disc loading time, **very** useful. **"File Copier"** allows **selective** copying from directory. **"Track and Sector"** a slower but very thorough program. Also includes the following useful utilities: **Viewbam, Fast Format, Unscratch, Disc Monitor, Menu Maker, Index, Retitle, Rename, Scratcher, Disc to Tape** and **Tidy**. If you have a 1541 you **must** have Disc Disector. Supplied on disc with full instructions.

ONLY £29.95

QUICKDISC
Incorporating Menu Maker
NOW EVEN FASTER

Speed up your 1541 disc drive with this **new** program. Loads most programs at **4 times** normal speed. A flexible package that allows you to incorporate a **menu and fast loading** on **your own discs**. Also includes a file exchanger for programs put on disc with **"Disco"** (**THE** tape to disc utility), another way of making fast load disc programs. A **must** for all disc drive users.

ONLY £11.95

FASTBACK
Fast Load Utility

Convert your slow loading software to **high speed load** with Fastback. The **most sophisticated** program of its type allows you to produce **"turbo"** loading versions of most single and **multipart** software that load **independantly** of Fastback. **Simple** to use **no programming skill needed**. Example: Load "The Hobbit" or "Manic Miner" in **150 secs**. Tried and trusted (over 2000 sold).

A very professional program.

ONLY £9.95

DISCO
THE TAPE TO DISC UTILITY

We **guarantee** that this is the **best** tape to disc utility on the market. **Very easy to use no programming knowledge needed**. Automatically handles device numbers and multi part programs. Programs **auto load** and **run** once transferred and can also be made to **fastload** with **Quickdisc**. Handy for making your own disc programs auto run.

ONLY £9.95

DOUBLER

Tape Back Up Device

Includes **hardware and software**. This **amazing** new cassette port adaptor allows you to make back up copies of **"turbo" software**. Tests have proved **100% successful**. Requires access to two cassette decks. The special software generates a **brand new machine copy** onto one deck while loading from the other. **Fast and reliable**, they said it couldn't be done.

ONLY £12.95

SPECTRUM

Yes, after spectacular success on the 64 Doubler is now available for the **Spectrum**. Tests have proved **100% successful.**

ONLY £12.95

PATENT APPLIED FOR

ROCKET

Fed up waiting for programs to load and save? Rocket is a utility for the programmer, speeds up loading and saving by about **8 times** while still allowing for normal speed operations. **Does not** affect the speed of BASIC as no wedge is used. Programs can also be saved with high speed system to **reload independantly**. A must for serious programmers who have no disc drive.

ONLY £7.95

BIGMOUTH

The **superb** new **speech synthesis** program for the '64. Uses "speak" as a basic command. **Very easy to use**. Can be incorporated into your basic programs. Words are entered in phonetic English, ie. spelt as they sound. Therefore vocabulary is **unlimited**. **Great fun** and good for educational purposes.

ONLY £7.95

COMMODORE CONNEXION

Connect **any** Centronics type printer to your 64. Price includes lead and software on cassette **with** special disc conversion program.

Simple to use and very reasonably priced

ONLY £14.95



3M SCOTCH DISCS

Lifetime guarantee Box of Ten

SS/DD **£16.00** DS/DD **£23.00**



DATA RECORDER

A **dedicated** data recorder designed for use on the CBM 64 or Vic 20. Specification as C2N but **with pause button.** Model available for the PLUS 4 or 16 at same price.

SATISFACTION GUARANTEED.
ONLY £29.95

All prices include P&P and VAT. Send cheque, Postal Order or Credit Card No. Credit Card orders accepted by phone. Callers welcome. Wide range of goods stocked. Trade enquiries welcome. European orders send price as advertised. Outside Europe £2.00 for airmail. Mail order to Evesham Shop please.

MICRO CENTRE
BRIDGE STREET, EVESHAM, WORCESTERSHIRE
Tel: 0386 49641

MICRO CENTRE
1756 PERSHORE ROAD, COTTERIDGE, BIRMINGHAM.
Tel: 021-458 4564

COMPUTER ASSISTED LEARNING

INFO-STREAM offers over 200 QUALITY EDUCATIONAL CASSETTES and DISKS for the Spectrum, BBC, Electron & CBM64, with SPECIAL OFFERS for the VIC 20 & ZX81.
Our new range covers VIDEO CASSETTES and BOOKS on BASIC, Assembly Language and Advanced Programming, together with intoductions to Forth, Lisp, Logo & Pascal.
Send 23p SAE for FREE INFORMATION PACK to: INFO-STREAM (Dept YC06), 36A Chase Green Avenue, Enfield, Middlesex, EN2 8EB. Export enquiries welcome.

496

Join The Commodore 64 Cassette Library

1. All the latest cassette based Software.
2. Only one price to hire: £1.50 per cassette, inc. p&p for full 7 days.
3. No charge on Hire of first game.
4. Membership charge £6.00 only.
5. Quick postal return.

Send cheque or postal order for £6.00 membership fee, or large SAE for further details to:

Commodore 64 Cassette Library
19 Darrell Way
Abingdon OX14 1HQ.
Tel. No. 0235 27769

529

VIC-20 USERS!

Cartridge and Cassette Backup. (5 ways) to tape or disc using the VICTRIX data booklet. Also contains information on adding RESET, SLOT POWER & WRITE PROTECT switches to your VIC-20.

Send £3.49 and 50p p&p
M. Parker, 52 Laurie Avenue, Forest Fields, Nottingham Ng8 1DY.
(Mail Order only please)

579

SINCLAIR QL
Only £379.99
QL COLOUR MONITOR
Only £255.95
COMMODORE SX-64 PORTABLE
(Disk Drive & Col. Monitor Built in) only £795.95
PHILIPS 12" Green Monitor (Composite video & Sound)
£85.95
SINCLAIR FLAT SCREEN TV
£99.95
All prices include VAT & Fast Free Delivery (UK only)
Please send cheques/PO's to: COMPUTER PLUS
45 Bridge Street, Newcastle. Staffs. ST5 1RT.
Tel. 0782 622333

580

• NOW AVAILABLE FOR THE AMSTRAD CPC 464!!

WIN THE POOLS?

SPECTADRAW 3 – THE LATEST VERSION OF THE ORIGINAL AND BEST POOLS PREDICTION PROGRAM FOR THE 48K ZX SPECTRUM!!

AND NOW – AMSTRA-DRAW – THE FIRST POOLS PREDICTION PROGRAM FOR THE AMAZING NEW AMSTRAD CPC 464!!



- Supplied with Database containing data on over 10,000 matches since 1980!
- You update the Database each week – but no tedious typing, as team and division names already in program!
- Errors easily corrected – the program even checks your entries!
- Comprehensive instruction manual and menu driven program – easy to use, even for a newcomer to computing!
- Will forecast the least likely draws for those who prefer to bet on fixed odds!
- Built in perm generator – complete your coupon direct from the screen!
- Fully microdrive compatible! (Spectadraw only)
- Compatible with Currah Microspeech – the first pools program to read you its predictions! (Spectadraw only)

Spectadraw 3 for the 48K Spectrum £9.95 inclusive
Amstra-Draw for the Amstrad CPC 464 £9.95 inclusive
(Cheques/PO's payable to B. S. McALLEY)

We dispatch every Monday with the database made up to include all matches up to the date of dispatch.

SPECTADRAW (Dept YC), 1 Cowleaze, Chinnor, Oxford OX9 4TD (Tel: 0844-52426)

530

SELL IT WITH YOUR COMPUTER

CLASSIFIED ORDER FORM

Classified Rates

Linage: 50p per word (Min 20 words) — prepayable

Linage advertisers should complete the form provided in BLOCK CAPITALS. Phone number counts as 2 words. Name and address to be paid for if used in advertisement. Box Number if required is £4.00 extra.

Display — rates per scc (Min 2scc)

One insertion	£9.00
Three insertions	£8.70
Six insertions	£8.50
Twelve insertions	£8.00

Display advertisers should provide separate copy and preferably reserve space by phone (01-661 3036).

Method of Payment

Cheques etc. should be made payable to Business Press International Ltd, and crossed. I enclose herewith cheque/PO for

£ ....

Please debit my Access/Visa Barclay Card/American Express/Diners Club Int/ /as below

SIGNATURE

Post to:

Cut out the order form and return together with your remittance to: Classified Department, Your Computer, Room H211, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS. (01) 661 3036

Please insert the following advertisement in Your Computer Classified Section

					LINAGE
					£10.00
					£12.50
					£15.00
					£17.50
					£20.00
					£22.50
					£25.00

No. of insertions required ☐ Box No. required YES/NO

NAME *(Please include initials)* ....

ADDRESS ....

....

....

Payment by credit card please state address card is registered

Daytime tel. no ....

THIS FORM SHOULD BE RETURNED BY MARCH 25th FOR MAY ISSUE PUBLICATION

Company Registered Number: 151537 (England) Registered Office: Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS.

Hot★ lines

Acorn Computers
Telephone: Cambridge 0223 210111

Atari
Telephone: Slough 0753 24561

Amstrad
Telephone: Brentwood 0277 228888

Commodore CBM
Telephone: Northampton 0536 205252

Enterprise
Telephone: 01-739 4282

Oric
Telephone: Teversham 02205 5141

Sinclair
Telephone: Camberley 0276 685311

Tandy Corporation
Telephone: Birmingham 021 5566101 ext 33

Toshiba
Telephone: Camberley 0276 62222

DATEBASE

Local, national and international micro events are updated here. To publicise your event, phone Paul Bond on 01-661 3472.

London Festival of Computing

Organised by ALCC and Inter-Action, together with the London New Technology Network, with GLC sponsorship. Takes place all over London from April 9 to April 20. Includes seminars for businessmen and a special venue for handicapped people at the Tara Hotel from April 14-19. Events all over London. Contact organisers on 01-240 8206, or write to 99 Longacre, WC2.

Microcity

Computers, Business Systems and Communications Equipment Exhibition takes place at the Bristol Exhibition Centre from May 14-16. Organised by Argus Specialist Exhibitions, 1 Park View, Berkhamsted, Hertfordshire. Telephone 04427 73291 for more details.

Northern Computer Show

Takes place at the Belle Vue Centre, Manchester from April 16-18. Business users only. Not open to the general public. More details from Reed Exhibitions, Surrey House, 1 Throwley Way, Sutton, Surrey. Telephone 01-643 8040.



Paul and David Cunningham were the two brothers to solve Melbourne House's Sherlock computer adventure game. They are pictured here with the runner up, who shot himself after receiving the complete range of Melbourne House books and software for the CBM-64.

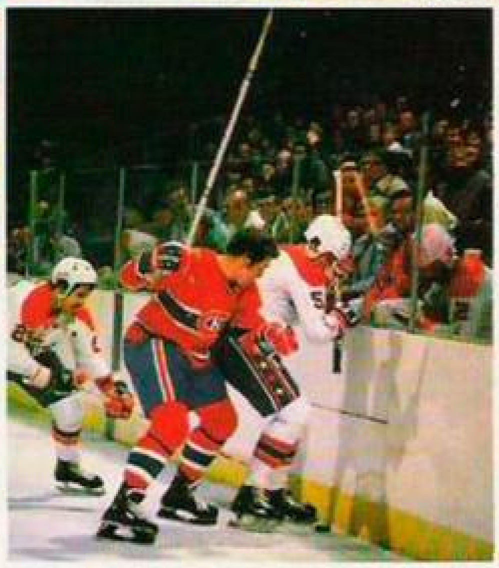
INDEX OF ADVERTISERS

FAST AND FURIOUS

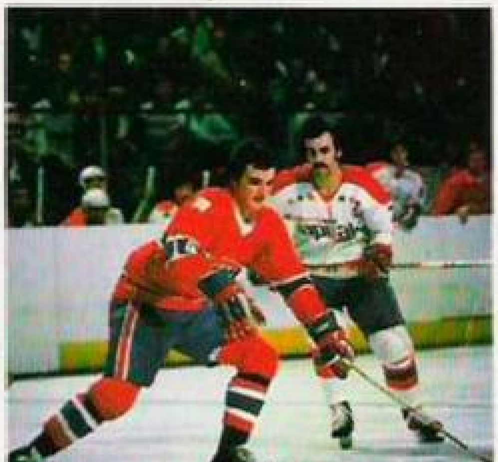
SLAP SHOT

ANIROG





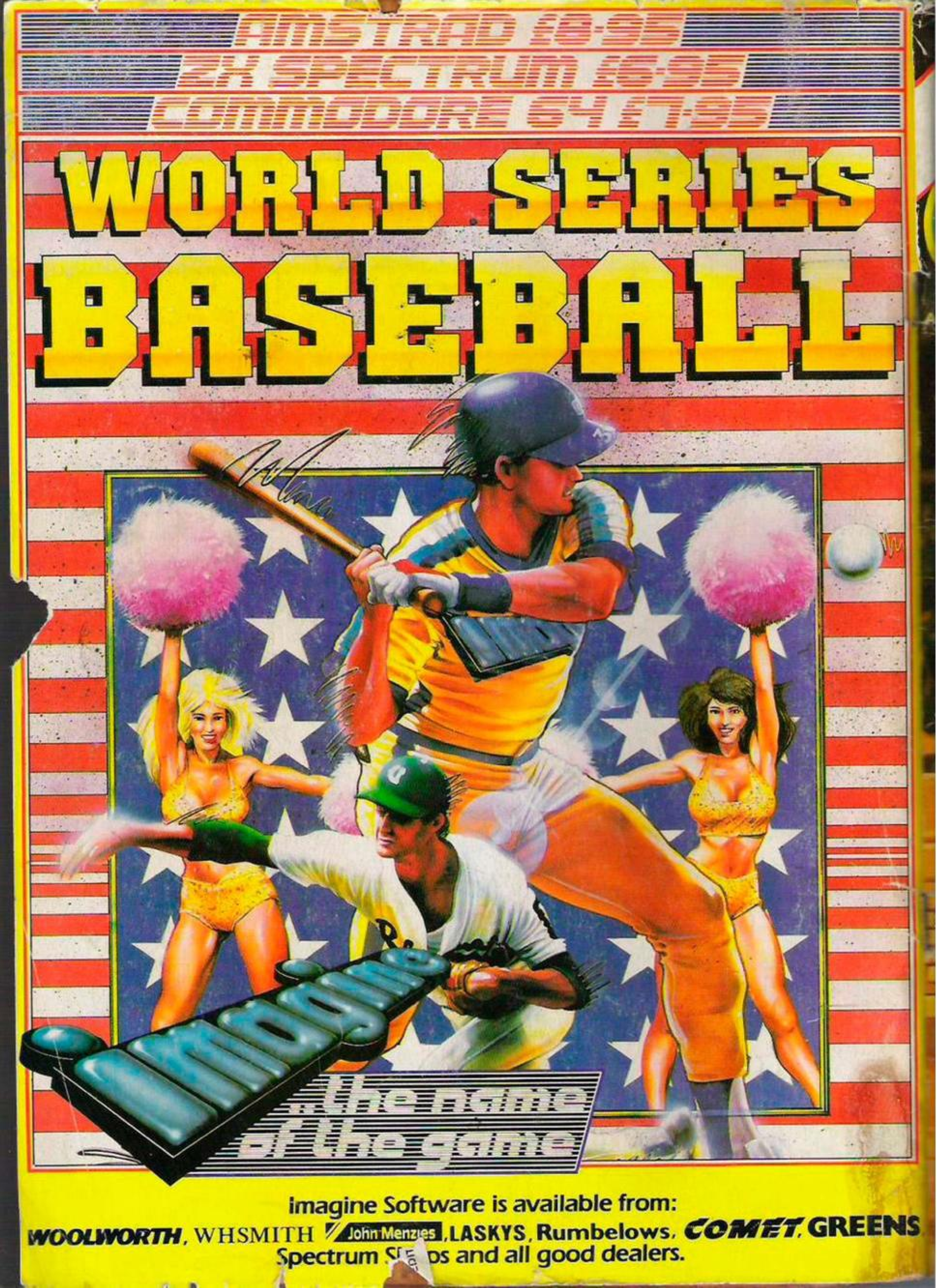
ALL ACTION ICE HOCKEY



SLAPSHOT from Anirog is a two player, fast action, ice hockey program. Also in this thrilling game there is included a smoothly running speech synthesis system of the very highest quality. Before the actual game starts each player must select an international hockey team to represent. After doing this you must quickly and skilfully manoeuvre your man across the glistening ice whilst your other team-mates move automatically around the rink waiting for you to pass the puck to them. If you do manage to race past the opposing defence then you have a chance to shoot for goal and score. However, you still have the goalie to beat who is able to dive in all directions. Also included in this amazingly quick and totally original game is the ability to actually physically bodycheck your opponents. Never the less, do not be too aggressive otherwise you could incur a "roughing penalty", resulting in a faceoff in your own end, giving the opposing side an easy chance of scoring. Also other extra features available include: Pause mode, and three levels of play ranging from fast to slow as well as a re-start option. Slapshot from Anirog is a totally original two player game with lightning fast action never seen before on the Commodore 64.

Two J.S. Commodore 64 Cassette £8.95 Disk £10.95

Trade Enquiries: ANIROG SOFTWARE LTD. UNIT 10 VICTORIA IND. PARK VICTORIA ROAD DARTFORD KENT (0322)92513/8
Mail Order: 8 HIGH STREET HORLEY SURREY 24 HOUR CR[illegible] SALES HORLEY 02934 6083



AMSTRAD £8·95
ZX SPECTRUM £6·95
COMMODORE 64 £7·95
WORLD SERIES BASEBALL
Imagine
..the name of the game
Imagine Software is available from:
WOOLWORTH, WHSMITH, John Menzies, LASKYS, Rumbelows, COMET, GREENS
Spectrum S ps and all good dealers.